Dual-hand detection for human-robot interaction by a parallel network based on hand detection and body pose estimation

Qing Gao, Jinguo Liu, *Senior Member*, Zhaojie Ju, *Senior Member*, and Xin Zhang

***Abstract*—In this study, a parallel network based on hand detection and body pose estimation is proposed to detect and distinguish human's right and left hands. The network is employed to human-robot interaction (HRI) based on hand gestures. This method fully uses hand feature information and hand information in the human body structure. One channel in the network uses a ResNet-Inception-Single Shot MultiBox Detector to extract hand feature information for human's hand detection. The other channel estimates human body pose first and then estimates the positions of the left and right hands using the forward kinematic tree of the human skeleton structure. Thereafter, the results of the two channels are fused. In the fusion module, the human body structure can be utilized to correct hand detection results and distinguish between the right and left hands. Experimental results verify that the parallel deep neural network can effectively improve the accuracy of hand detection and distinguish between the right and left hands effectively. This method is also used for the hand gesture-based interaction between astronauts and an astronaut assistant robot. Our method can be suitably used in this HRI system.**

***Index Terms*—Assistant assistant robot, Dual-hand detection, Human body estimation, Human-robot interaction, Parallel deep neural network**

I. Introduction

HUMAN-robot interaction (HRI) is important in robotics and aims to make humans and robots to communicate with each other. Robots have smooth operation, high precision, and wide range, whereas contrast, humans can make perceptions, decisions, and plans efficiently. Therefore, if they can work collaboratively, then they will finish tasks effectively.

Traditional HRI methods are centered on robots. The development of technologies, such as computers and artificial intelligence, has gradually evolved HRI from robot-centered to human-centered methods. These new HRI methods are based primarily on hand gestures [1], voice [2], and electroencephalogram [3]. These methods are the main directions for the future development of HRI because of their natural and intuitive features.

Among these methods, the vision-based hand gesture HRI is a good choice [4]–[6]. It includes hand detection, hand gesture recognition, and hand tracking. Hand detection is the premise and basis for the others. Vision-based hand detection is a special object detection. But it has a few disadvantages, such as complex background, occlusions, and illumination variation. These factors influent the precision of hand detection and recognition. Traditional visual-based hand detection methods are based primarily on skin color, motion flow information, and shape models [7]. These traditional methods only extract the shallow information of the hand, and they are subject to some restrictions. For example, skin color-based hand detection relies heavily on skin color information. Motion flow-based hand detection cannot detect static hands. Moreover, model-based hand detection is heavily influenced by complex backgrounds.

Compared with traditional hand detection methods, deep learning method can extract deep abstract hand features, and minimize the disadvantages of vision-based hand detection. Therefore, we can also use deep learning models [8]–[10] that are used for object detection. Some of these models, such as region-based convolution neural network (R-CNN) [11], Fast RCNN [12], Faster RCNN [13], you only look once (YOLO) [14], and Single Shot MultiBox Detector (SSD) [15], have achieved great effects. However, false detection or detection failure occurs in the hand detection process even when these deep learning models are used.

Distinction between the left and right hands is more difficult than hand detection especially when too many hand gestures are available, because both hands are only slightly different. Reference [16] uses the assumption that defines the direction and position of both hands to solve the problem of hand distinction in hand tracking. Nonetheless, if the hand direction or position is not in the defined condition, then this method will not work. Reference [17] and [18] use deep learning model to detect and distinguish the right and left hands. However, this method is only suited to several special hand gestures, such as the hand gestures used in driving vehicles. If too many hand gestures are available, then the use of only deep learning models to detect hand features for distinguishing the right and left hands is insufficient.

Hand detection is distinct from the detection of other objects because most objects are independent, but hands are dependent. Hands are related to the human body. When we observe other people's hands, we base not only the characteristics of the hand, but also the structural features of the human body. Therefore, this study simulates the way humans detect hands. We combine the hand detection based on hand features with the dual-hand position estimation in accordance with the hu-

man body structure to detect human's left and right hands. As a result, a parallel deep neural network is designed. First, the two sub-networks connected in parallel uses the ResNet-Inception-Single Shot MultiBox Detector (RI-SSD) and human pose estimation method to extract the characteristics of the hand and human body structure, respectively. Then, a fusion module is used to fuse the results for obtaining human's dual-hand detection. To the best of our knowledge, we are the first to combine the hand detection with human body estimation for dual-hand detection.

We summarize the main contributions of our work as follows: (1)a RI-SSD network that changes the structure of SSD using ResNet and Inception network is proposed to increase the accuracy of hand detection. (2)An improved body pose estimation method based on body structure forward kinematic (FK) tree is proposed to estimate the keypoints of left and right hands. (3)A fusion module is proposed to fuse the results of hand detection and dual-hand position estimation for dual-hand detection. (4)Our dual-hand detection method can be used in static and dynamic hand gesture recognition systems. The method is also used in a space HRI system.

The remainder of the paper is structured as follows. In Section 2, the RI-SSD structure is introduced. In Section 3, the body pose estimation and dual-hand detection method with fused information are introduced. Experimental results and validation are presented in Section 4. Section 5 elaborates the conclusion and future work.

II. RI-SSD for Hand Detection

In the hand detection channel, the deep learning method is used to detect human's hands. SSD [15], which is proposed by WeiLiu et al., is a good choice for object detection. Its accuracy rate is higher than that of YOLO [14], and its speed is faster than that of Fast-RCNN [12]. The SSD method is based on the anchor of Faster R-CNN. As shown in Fig.1(a), SSD uses the traditional classification network VGG-16 and introduces some additional layers as feature extraction layers. The changes in the extra layer ratios are evident. Thus, SSD method can detect multi-scale objects.

However, performance of SSD in detecting small targets is unsatisfactory. The reason is that the shallow layers of this method have sufficient contextual information but inadequate semantic information, whereas the deep layers have sufficient semantic information but inadequate contextual information. The two kinds of information are essential for detecting small objects. When humans interact with robots, a long distance may exist between them. At this time, the human's hands are regarded as small targets. These small targets need sufficient contextual information to provide detailed features and intensive sampling and sufficient semantic meaning to distinguish them from the background [19]. Consequently, use of the SSD alone to detect human's hand directly is insufficient. To solve the problem of SSD, we propose an improved SSD network called RI-SSD.

We redesign the SSD structure, inspired by GoogleNet's Inception block [20] and the deep residual network [21], to ensure accuracy in object detection, especially for small

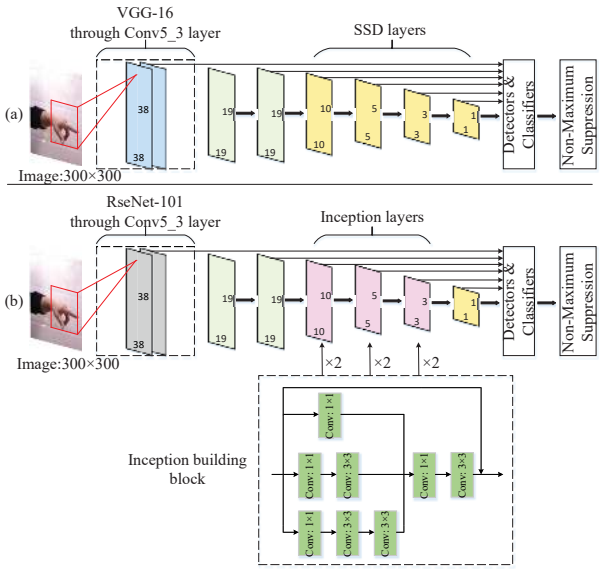

Fig. 1. (a) The structure of SSD network (b) The structure of RI-SSD network.

objects. First, we use the ResNet101 network [21] instead of the VGG-16 of the SSD. Compared with the VGG-16, the ResNet101 can extract image features better, which helps to improve the detection rate of hands. Second, the Inception structure is added to the deep layers of the SSD. The deeper the network layer is, the more powerful its abstract features become. However, some training problems, such as gradient disappearance and over-fitting, are also generated. Considering the tradeoff between performance and speed, we introduce the Inception structure in some extra layers behind ResNet101 to increase the types of convolution kernels. Thus, the scope of the receptive field is expanded. Thereby increasing the sensitivity of the model to small objects and avoiding loss of large objects.

Figure 1(b) shows the structure of RI-SSD. It includes 13 layers. The first five layers are the first five layers of RseNet. The sixth and seventh layers are convolution layers changed by fully connected (FC) layers of RseNet, the following three layers are Inception layers, and then a 1×1 sized SSD layer follows them. The last two layers are FC and classification layers. Each layer except the last two layers has convolution, pooling and rectified linear unit (ReLU) layers. The input layer size is 300×300, and the other feature maps' sizes are shown in Figure 1(b). The traditional SSD uses the VGG-16 for feature extraction in shallow layers (light blue box in Figure 1(a)). According to the reference [22], the image feature extraction effect of ResNet-101 is better than that of VGG-16; therefore, the VGG-16 layers in the SSD are replaced by the ResNet-101 layers (gray box in Figure 1(b)). The traditional SSD uses deep layers to capture objects, and the deep SSD layers use only one type of convolution kernel (a 3×3 convolution kernel). Through the convolution operation, deep feature maps will produce the location offset and confidence of the object. In object detection, large convolution kernels capture large objects, and small receptive fields can locate

small objects. Therefore, the feature maps in shallow layers may lose the details of objects. Accordingly, we modify some SSD layers and replace these layers with the Inception building block (green box in Figure 1(b)) to form the Inception layers (purple box in Figure 1(b)). A 1×1 convolutional kernel, a 3×3 convolutional kernel, and a 5×5 convolutional kernel are stacked in the Inception building block instead of the original 3×3 convolutional layer. The 5×5 convolution kernel is replaced by a series of two 3×3 convolution kernels. In this way, considerable details of objects can be obtained. We reduce the number of feature maps for each layer on the Inception building block to match the total number of original feature maps. In reflecting different proportions of the receptive fields, we set different weights ($w = 1, 2, 1$) to the output of each type of convolution (conv1×1, conv3×3, conv5×5). In this approach, the network can capture large and small objects effectively.

The network outputs the confidence and location of human's hands. Its loss function is [15]:

$$f_h(x_{ij}^{hand}, c_h, l, g) = \frac{1}{N}(f_{h\_c}(x_{ij}^{hand}, c_h) + \alpha f_{h\_l}(x_{ij}^{hand}, l, g)) \tag{1}$$

$$f_{h\_l} = \sum_{i \in Pos}^{N} \sum_{hand \in (x,y,\omega,h)} x_{ij}^{hand} smooth_{L1}(l_i^{hand} - \hat{g}_j^{hand}) \tag{2}$$

$$f_{h\_c} = -\sum_{i \in Pos}^{N} x_{ij}^{hand} log(\hat{c}_{ih}) - \sum_{i \in Neg} log(\hat{c}_{i0}) \tag{3}$$

The loss function f_h includes two parts, namely, confidence and location losses. In the equation above, *N* is the number of default boxes for matching with ground truth box. N is set to 4. Because in the reference[15], the N is set to 4, in order to compare our method with the SSD, we set the same value of N. $x_{ij}^{hand} = (0, 1)$ indicates the *i*-th hand default box matches the *j*-th hand ground truth box. c_h is hand confidence. l is prediction box, and *g* is groundtruth box. Parameter α is used to adjust the ratio between confidence and location losses, and α is set to 1 to balance the weights of these losses. f_{h_c} is confidence loss which uses softmax loss. $i \in Neg$ means there is no hand in the i-th default box. $\hat{c}_{ih}$ means hand confidence of i-th default box, $\hat{c}_{i0}$ means background condifence of i-th default box. When the *i*-th default box matches the *j*-th ground truth box of hand category, the higher the probability of the hand, the smaller the loss. When there is no hand in the *i*-th default box, the higher the probability of background, the smaller the loss. f_{h_l} is the location loss which uses smooth $L1$ loss. The (x, y, ω, h) are the centre position (x, y), width (ω) and height (h) of the hand groundtruth box. $i \in Pos$ means the i-th default box matches a ground truth box. l_i^{hand} means the i-th prediction box, $\hat{g}_j^{hand}$ means the j-th groundtruth.

III. Dual-hand detection method

A. Parallel deep neural network structure

In detecting the hands in real life, we consider not only the characteristics of the hand, but also the characteristics of the hand in the human body structure. This approach enables accurate hand detection. At present, most methods of hand detection are only founded on the characteristics of the hand. As a result, the detection fails or localization errors occur when the background is complex or the hand is partially occluded.

Accordingly, this study simulates the way humans detect hands. The characteristics of the hands and the hands in the human body structure are combined to detect the human's pair of hands. A parallel deep neural network structure is designed. One sub-network adopts the above-mentioned RI-SSD to detect hands by extracting the characteristics of hands. The other sub-network estimates dual-hand positions by extracting the characteristics of the human body structure. Thereafter, a fusion module is utilized to fuse the results of the two sub-networks to detect human's two hands.

In this process, the hand image is input to the hand detection sub-network and the human pose estimation sub-network. The hand detection sub-network detects human's hand by the RI-SSD network. In this process, the location of a box surrounding the hand and the confidence of the hand can be obtained. Meanwhile, the human pose estimation sub-network estimates human body pose and obtains the body skeleton. Then, the human body forward kinematics (FK) tree is utilized to obtain the positions of both hands. Finally, the outputs of the two sub-networks are merged through the result fusion module to obtain the location and confidence of the box surrounding the left and right hands.

B. Dual-hand position estimation based on body pose estimation

In the human pose estimation sub-network, we propose an improved human pose estimation method which can estimate the central positions of human's left hand right hands based on an existing pose estimation method and human body forward kinematics tree. In this part, the human body pose should be estimated first. The human pose estimation method in [23], [24] is used. This method uses part affinity fields (PAF) to achieve 2D pose estimation for multiple people, and ensure real-time performance of pose estimation. The human body pose estimation database uses the COCO Keypoints Challenge database. A total of 18 keypoints, which correspond to human joints, are available in the database. The keypoints near the hand positions are the left and right wrists. Our goal is to estimate the center positions of the right and left hands. Thus, we need to use the human forward kinematics to estimate the positions of both hands. The method framework is presented in Figure 2.

As shown in Figure 2, the human pose estimation method first uses the VGG-19 to obtain the astronaut image features, and then passes the image features into part conference map (PCM) [23] and PAFs to obtain the human pose based on the COCO human body keypoints. Thereafter, we obtain the left hand center position L_l and the right hand center position L_r through the human body FK tree module. The output of Stage1 is the corresponding PCM map $S1$ and PAFs map $L1$. The input of Stage2 includes the outputs $S1$ and $L1$ of Stage1 and the feature map of the original image. Notably,

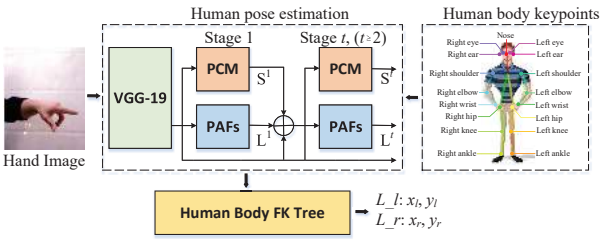

Fig. 2. Human pose estimation framework.

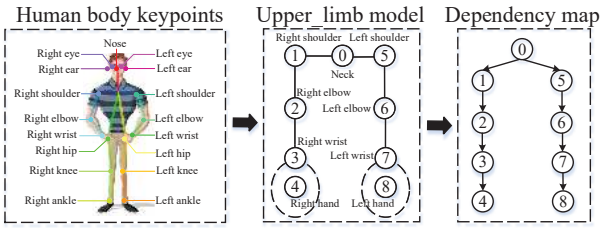

Fig. 3. Dependency map.

the network at each subsequent stage is similar to Stage2. The loss function of the model is the norm $L2$ which describes the distance between the prediction result and the ground truth [24]:

$$f_S^t = \sum_{j=1}^{J} \sum_{p} W(p) \cdot \left\| S_j^t(p) - S_j^*(p) \right\|_2^2 \tag{4}$$

$$f_L^t = \sum_{c=1}^{C} \sum_{p} W(p) \cdot \left\| L_c^t(p) - L_c^*(p) \right\|_2^2 \tag{5}$$

where $S = (S_1, S_2, \cdots, S_J)$ has J confidence maps; $L = (L_1, L_2, \cdots, L_C)$ has C vector fields , f_S^t and f_L^t are the loss functions of the output S^t and L^t of the Stage t, respectively, S_j^t is the output PCM map of Stage t, L_c^t is the output PAFs map of Stage t, S_j^* is the ground truth of PCM, L_c^* is the ground truth of PAFs, W is a binary mask, p is an image position. Then, the final loss function of the network f is

$$f = \sum_{t=1}^{T} (f_S^t + f_L^t) \tag{6}$$

T is the number of the Stage, and it is set to 6. Because based on the reference[24], in Stage 6, it can get both high accuracy and fast speed. Human body keypoints are obtained from the COCO Keypoints Challenge database, which has a total of 18 keypoints, as shown on the right side of Figure 2. However, the keypoints selected at the positions of the hands are the left and right wrists (most of the human body database keypoints select the left and right wrists). Therefore, the output of human pose estimation should be input into the human body FK tree module to estimate the center positions of human's two hands [25].

Prior to inputting the human body pose estimation result into the human FK tree module, the dependency map including the right and left hands from the human body keypoint map must be extracted first. As shown in Figure 3, the human upper limb model is first extracted from the human keypoint map and numbered. Then, the dependency map with parent and child nodes is set up accordance with the numbers. Finally, on the basis of the dependency of the parent and child nodes, the center positions of the left and right hands are estimated.

I denotes an image containing an astronaut, and $p_i = (x, y)$ denotes the pixel position of the i-th keypoint in the image, where $i \in \{0, 1, 2, \cdots, 8\}$. The keypoints correspond to the human' upper limb joint points, and the edges (i, j) of each pair of adjacent nodes in the map represent the following dependencies:

$$L_j = s \cdot h_{i,j}(L_i, I, s) + L_i \tag{7}$$

where i and j are a pair of parent and child nodes in the dependency map, and s is the scale parameter. $h_{i,j}$ is a function with a two-dimensional vector as the output that represents the relative positions of the parent and child nodes. Have regard to the position of root node L_i, the scale s and an image I, we can refer to the dependency map and estimate the positions of the left and right hands (L_4 and L_8) by Equation (5).

The function $h_{i,j}$ in Eauation (5) is defined as follows:

$$h_{i,j}(L_i, I, s) = g_{i,j}(f(L_i, i, s)) \tag{8}$$

where $g_{i,j}$ is a regression. $f(L_i, i, s)$ is a predefined function that computes the image features of the image block centered at L_i in scale s. The size of the image block is sufficiently large to contain as little background as possible while including all possible L_j.

Each $g_{i,j}$ is a multidimensional output that generates a two-dimensional vector. A parent node i may have multiple child nodes $\{j_1, j_2, \cdots, j_L\}$. Given that the input characteristics are the same, we can define a multidimensional output regression to output the relative positions of all child nodes:

$$g_i(\cdot) = (g_{i,j_1}(\cdot), \cdots, g_{i,j_L}(\cdot)) \in R^{2L} \tag{9}$$

Therefore, the left wrist position L_3 and right wrist position L_7 can be obtained by the human body pose estimation method, where the left wrist is regarded as the root node of the left hand, and the right wrist is regarded as the root node of the right hand. Thus, the center positions of human's two hands L_4 and L_8 can be obtained using Equations (5)-(7).

C. Dual-hand detection with fused information

The hand detection sub-network can detect hand and locate the range of hand. However, the method cannot distinguish between the left and right hands and results in false detection in some cases. The human body pose estimation sub-network can distinguish between the left and right hands and estimate the center position of hands. Nevertheless, the approach produces inaccurate estimated positions and cannot locate the range of hands. The fusion module can combine the output results of the two sub-networks to realize the detection and

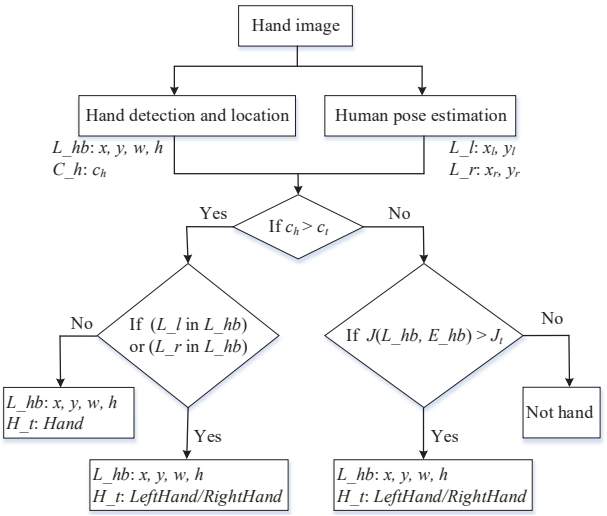

Fig. 4. Fusion method flow chart.

precise localization of human's left and right hands, reasonably and effectively. The fusion module is shown in Fig.4.

In Figure 4, the L_hb indicates the location of the detected hand box, x, y are the center coordinate values of the hand box, w and h are the width and height of the hand box, respectively, L_l and L_r are the positions of the estimated right and left hands L_4 and L_8, respectively, E_hb is the location of the estimated hand box, and H_t indicates the type of hand. The hand is divided into three types. H indicates the hand, LH indicates the left hand, and RH indicates the right hand. Whether the hand confidence c_h output by the hand detection sub-network is greater than a threshold confidence c_t is determined. If c_h is greater than the threshold c_t, then the hand detection result is completely trusted.

At this time, the hand position estimated by the human body structure is only used to assist in distinguishing the right and left hands. That is, whether the estimated position of hand is in the hand box area is determined. If the position of the left hand is in the hand box, then the area is deemed to be the left hand. If the position of the right hand is in the hand box, then the area is considered to be the right hand. If the positions of the left and right hands are not in the hand box, then the area is only displayed as a hand. The judgment formula is

$$\begin{cases} LH & \text{if } (x - \frac{2}{w} < x_l < x + \frac{2}{w}) \cap (y - \frac{2}{h} < y_l < y + \frac{2}{h}) \\ RH & \text{if } (x - \frac{2}{w} < x_r < x + \frac{2}{w}) \cap (y - \frac{2}{h} < y_r < y + \frac{2}{h}) \\ H & \text{else} \end{cases} \tag{10}$$

If c_h is smaller than the threshold c_t, then the estimated positions of the left and right hands are used to determine and correct the hand detection and location results. That is, the estimated hand box, E_hb, is attracted by the estimated hand position. As shown in Figure 5, the box is centered on the position of the hand estimated in accordance with the human body structure, and the side length of the square l is the distance between the wrist and the corresponding hand. Then, we calculate the Jaccard similarity of E_hb and L_hb

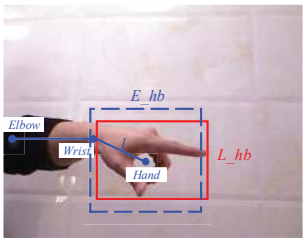

Fig. 5. L_hb and E_hb.

by the hand detection and location sub-network. The Jaccard similarity calculation formula is shown as follows [15]:

$$J(L\_hb, E\_hb) = \frac{|L\_hb \bigcap E\_hb|}{|L\_hb \bigcup E\_hb|} \in [0, 1] \tag{11}$$

J_t is define as the threshold of Jaccard similarity value. If the similarity is greater than J_t, then the L_hb area is considered to be a hand, and the left and right hands are distinguished by the hand estimation result. If the similarity is less than or equal to J_t, then the L_hb area is considered not to be a hand. The judgment formula is

$$\begin{cases} LH/RH & \text{if } J(L\_hb, E\_hb) > J_t \\ Nothand & \text{if } J(L\_hb, E\_hb) \leq J_t \end{cases} \tag{12}$$

J_t is chosen as 0.5. Because through experiments, when the J_t is 0.5, it can get the highest accuracy for dual-hand detection.

IV. Experimental Results and Validation

A. Validation of RI-SSD

For the hand detection database, we select two public databases, namely, Oxford hand database [26] generated by Oxford University and the Egohands database [27] produced by Indiana University. The Oxford hand database is collected from various public image dataset sources, such as Skin and 2007 and 2010 PASCAL VOC datasets. Most images are random hand gestures in our daily life. The Egohands database contains 48 different videos of egocentric interactions with pixel-level ground-truth annotations. Most images are hand gestures that interact with objects. We train the SSD network on both databases. We find that we can obtain 96.64% accuracy rate on the Egohands database. However, the hand gestures of Egohands are those for catching objects. We cannot achieve good performance on other hand gestures. Meanwhile, we can only obtain 68.74% accuracy rate on the Oxford hand database, which includes many types of hand gestures in daily life. Thus, we combine the training images of the two hand databases and train the SSD and our RI-SSD on this combined database.As a result, the final number of training images is 7029.

The experiment is carried out in the Caffe environment, and the SSD and RI-SSD are tested on the testing images of Egohands database, Oxford hand database and their combined database. In the training stage, we use stochastic gradient

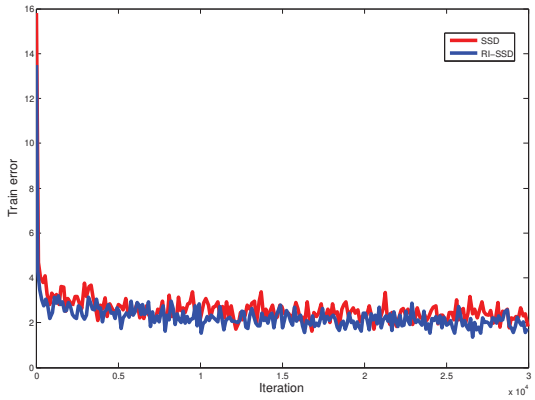

Fig. 6. The Training error curves of SSD and RI-SSD.

TABLE I
MAP AND SPEED OF SSD AND RI-SSD

Model name	Test database	mAP	Speed(ms)
SSD300	Oxford hand database	74.16	9
	Egohands database	97.03	
	Oxford+Egohands	85.60	
RI-SSD300	Oxford hand database	80.21	17
	Egohands database	97.56	
	Oxford+Egohands	88.89	

descent (SGD) method. The initial learning rate, momentum, and weight decay are set to 0.001, 0.9, and 0.9, respectively. The change mode of learning rate uses the multistep method. The total number of training sessions is 60000 iterations. The learning rate successively drops by 10 at 20000 and 40000 iterations. The train and test processes are conducted under the GTX 1060 GPU.

The training error curves of SSD and RI-SSD are shown in Figure 6. The test mean average precision (mAP) and speed of the two networks are shown in Table I.

As shown in Figure 6 and Table I, when the test database is combined with Oxford hand and Egohands databases, the accuracy of the RI-SSD300 is 3.29% higher than that of the SSD300, and the RI-SSD300 can still achieve real-time hand detection. Therefore, the proposed RI-SSD network structure can effectively improve the accuracy of human's hand detection. This result is due to that the Oxford hand database has many small-sized hands, and the RI-SSD can detect these hands better than SSD network.

The detection effects of different hand gestures in the Oxford hand database are shown in Figure 7 for demonstrating the effectiveness of the RI-SSD further. As shown in Figure 7, the designed RI-SSD can get a better detection accuracy than that of the SSD network on small hand images.

In Figure 7, the first row contains the results from the SSD300, and the second row includes the results from the RI-SSD300. Hands in images are circled with bounding boxes. Evidently, Figure 7 demonstrates that the designed RI-SSD can obtain better detection accuracy than SSD network on small hand images.

Fig. 7. Hand detection of SSD and RI-SSD on some images [26].

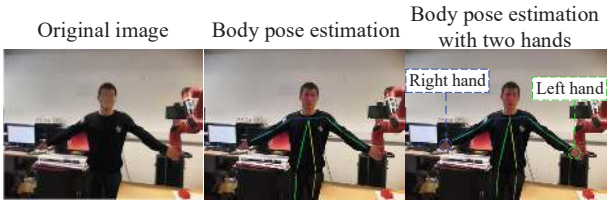

Fig. 8. The results of estimated dual-hand positions.

B. Body pose estimation with dual-hand positions

Center positions of human's two hands are estimated on the basis of the above-mentioned hand position estimation principle. The specific steps are described as follows:

1) Estimate the astronaut's human skeleton structure in accordance with the human body pose estimation method.
2) Extract the dependence map of the human upper limb joint points including the center points of the two hands.
3) Estimate the positions of human's two hands in accordance with the human body KF tree.
4) Select the center points as the final left and right hand positions within the range of estimated position points.

The estimated results are shown in Figure 8, where the left image is the original image, the middle image is the body pose estimation using the method of reference [24], and the right image is the body pose estimation with the positions of the left and right hands by the human body structure FK tree. In this image, the right hand position is indicated by a blue dot, and the left hand position is indicated by a green dot. Figure 8 shows that the method can estimate the positions of human's two hands effectively.

C. Dual-hand detection based on RI-SSD and body pose estimation

During the experiment of human hand detection, the detection of the RI-SSD has shown several disadvantages.

1) The method cannot distinguish between human's left and right hands.
2) Some objects in the background that are similar to a hand in color or shape will be mistaken as hands when the background is complex.
3) A location error or a detection failure may occur when the hand size in the image is small or the hand image is blurred due to excessive speed.

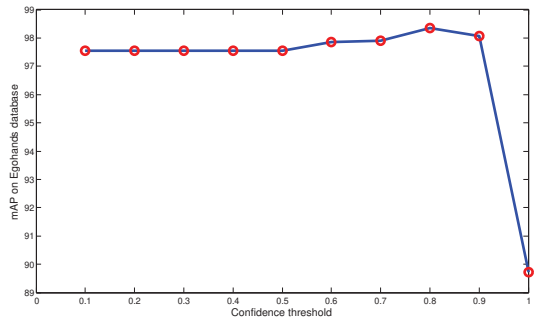

Fig. 9. mAP line of hand detection.

TABLE II
MAP OF RI-SSD AND THE PARALLEL NETWORK

Model name	mAP of hands	mAP of left hands	mAP of right hands
RI-SSD	97.56	–	–
Parallel network	98.34	89.27	90.18

The fusion method of the parallel network mentioned above demonstrated that the hand detection results output by the RI-SSD network and the dual-hand positions estimated in accordance with the human body pose can be effectively combined to distinguish human's left and right hands. Hand detection errors can also be avoided.

Labels of Oxford hand database are only hands. The database is not appropriate for distinguishing the left and right hands. The Egohands database labels have the left and right hands. Therefore, we use the Egohands database in this experiment.

For the confidence threshold c_t, we select the best value of c_t through experimentation. The value is changed from 0.1 to 1, and the mAP line of the hand detection using our parallel deep neural network is shown in Figure 9.

Accordingly, we use 0.8 as the value of c_t. We also test the RI-SSD and our dual-hand detection method using the parallel deep neural network on the Egohands database. The test results are given in Table II.

Figure 9 and Table II show that the parallel deep neural network structure, which combines the hand detection and location results and the human body pose estimation results, is more accurate than the RI-SSD that extracts only the hand features. The proposed method is also effective in distinguishing between the right and left hands. In addition, the detection accuracies of the left and right hands are lower than the hand detection accuracy. This result is due to that some volunteers' hands are too close to the camera during the acquisition process. Hence, the area of the hand in the image becomes too large to estimate the pose of the human body. As a result, the positions of the left and right hands cannot be estimated and can only be identified as hands.

In order to compare our method with the state-of-the-art dual-hand detection methods, we test our method on the VIVA Hand Detection database. It's a public dual-hand detection database, and it consists of driving car hand gestures from 54 RGB videos collected in naturalistic driving settings of illumination variation, large hand movements, and common occlusion. The result is shown in Table III.

TABLE III
DUAL-HAND DETECTION RESULTS ON THE VIVA HAND DETECTION DATABASE

Model name	Static only	mAP of left hands	mAP of right hands
MS-RFCN [17]	Yes	75.3	69.8
Ours	Yes	64.2	59.6
CNN with spatial region sampling [27]	Yes	52.7	42.3
ACF [28]	Yes	47.5	33.7
Modified Faster-rcnn	Yes	39.0	12.0

Fig. 10. For image 1, the RI-SSD and parallel network can both detect and locate both hands, but the parallel network can also distinguish between the right and left hands. For image 2, when the hands are too large foe estimating the human body pose, the parallel network can only detect the two hands as a hand. For image 3, RI-SSD generates an incorrect detection, but the parallel network can remove this incorrect detection.

Table III shows that our method can get a good result on the VIVA database. The result of our method is better than most of the methods but not the best. For one thing, our method relies on body pose estimation. Some images in VIVA database have a little body information, so our method cannot distinguish between left hand right hands. For another, the advantage of our method is detecting dual hands for a variety of hand gestures. But the VIVA database only contains a few gestures like holding the steering wheel gesture, the result on this database does not fully demonstrate the superiority of our method.

Some experimental results are shown in Figure 10, where the three images in the first column are the original images, the three images in the second column are the results of RI-SSD, and the three images in the third column are the results of the parallel deep neural network. The range of the right hand in the image is marked by a blue box, the range of the left hand in the image is marked by a green box, and the range of the hand in the image is marked by a red box. Figure 10 shows that the proposed fusion method and the parallel deep neural network can distinguish and locate the locations of the left and right hands effectively.

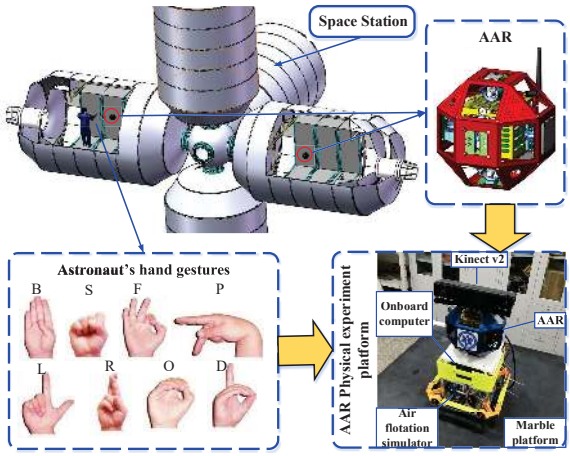

Fig. 11. AAR-2 platform.

D. Application in astronaut-robot interaction system

1) Second generation astronaut assistant robot (AAR-2) platform: Second generation astronaut assistant robot (AAR-2) is an in-cabin flying robot used in space stations [29], [30]. It has high intelligent, and astronauts can communicate with it face-to-face using hand gestures. It can also collect information from the space station to the astronauts to assist them completing some space tasks. The upper part of Figure 11 shows an imaginary map of an astronaut interacting with the AAR-2 through hand gestures in the space station. During the gesture-based interaction between the astronaut and the AAR-2, the astronaut uses hand gestures to convey instructions to the AAR-2. The AAR-2 collects hand gesture images through a vision sensor (Kinect v2), and then detects and recognizes the astronaut's hand gestures. Thereafter, recognized hand gestures are converted into instructions, through which the AAR-2 can perform the corresponding operations [1]. AAR-2's physics experimental platform is illustrated in the bottom right of Figure 11. The AAR-2 is mounted on an air float simulator and can simulate space microgravity environment and help the AAR-2 to move in three degrees of freedom (translational motions along x and y axises and rotational motion around z axis) on a marble platform. At present, the AAR-2 can realize functions such as translational motion, rotational motion, target approximation and data transmission.

When astronauts interact with AAR-2 face to face using hand gestures, they need natural and reasonable hand gestures. For astronaut's SHRI hand gestures, the following requirements must be met:

a) Hand gestures should be simple and easy to learn.
b) Hand gestures should be natural and reasonable.
c) Hand gestures can be used to not only send control commands to the AAR-2 but also control the real-time motion state of the AAR-2.

On the basis of these requirements, a dual-hand gesture that uses the left hand to send commands and the right hand to operate the AAR-2 is designed. The left hand of the astronaut is primarily used to send control commands to the AAR-2. For these hand gestures, we chose the American signal language (ASL) hand gesture database [31]. This database contains 26 hand gestures that represent 26 English letters. It is appropriate to SHRI because it can be memorized easily and is reasonable and natural. Eight hand gestures of the ASL are selected as the left-hand gestures, which are shown in the bottom right of Figure 11. The ASL letter corresponding to each hand gesture is the first letter of the control command. The astronaut's right hand is utilized to manipulate the motion state of the AAR-2. For example, when the left-hand gesture is "Line motion", the AAR-2 performs translational motion by detecting the translation of the right hand. When the left-hand gesture is "Rotational motion", the AAR-2 performs a rotational motion of roll, pitch, or yaw by detecting the rotation of the right hand.

TABLE IV
MAPS OF DIFFERENT HAND GESTURES ON SRSSL DATABASE

Model name	B	S	F	P	L	R	O	D
Our method	97.35	98.63	96.56	98.46	98.48	97.94	98.95	96.97

TABLE V
MAPS OF LEFT HAND AND RIGHT HAND ON SRSSL DATABASE

Model name	mAP of left hands	mAP of right hands
Our method	85.66	85.97

2) Space Robot Simple Sign Language (SRSSL) database: The database of astronauts' hand detection and location uses the self-made Space Robot Simple Sign Language (SRSSL) database. This hand gesture database collects the RGB image hand gestures from six volunteers, and each of them include the eight astronaut-robot interaction hand gestures. The hand gesture images from each person have 100 images. That is, 600 hand gesture images are available. Hand gesture images from five people (500 images) are selected as the training data, and the other person's hand gesture images (100 images) are used as the test data.

Transfer Learning method [21] is employed in the training process to save time. We train the RI-SSD on this database. At the beginning, we train on the hand database synthesized by Oxford hand and Egohands databases. Thereafter, we retrain the network on SRSSL database by transfer learning method.

3) Experimental result: After the training, we combine the $RI-SSD_SpaceHandNet$ with the human body pose estimation method mentioned above and obtain the parallel deep neural network for the dual-hand detection of SRSSL database. The mAPs of different hand gestures on the SRSSL database are shown in Table III, and the mAPs of the right and left hands on the SRSSL database are shown in Table IV.

As shown in Tables III and IV, the proposed parallel network can detect the hand gestures in SRSSL effectively. It also can distinguish between the left and right hands accurately. The accuracies of the left and right hands are lower than the accuracies of these hand gestures. This result is due to that the SRSSL database has many large hand images and that the human pose cannot be estimated. Our method can

be implemented in the astronaut-robot interaction platform effectively.

V. Conclusion remark and future work

This paper presented a parallel deep neural network that combines the characteristics of the hand with those of the human body. This method can effectively improve the accuracy of hand detection and distinguish between the left and right hands. Moreover, this method can be implemented to the interaction between the astronaut and AAR-2 to realize the detection of astronauts' two hands.

The contributions of the study are summarized as follows: (1)A parallel deep neural network structure was designed. This network can extract the characteristics of the hand and human body structure. It can also effectively detect and locate the left and right hands of astronauts. (2)In the hand detection sub-network, a RI-SSD was designed. The designed RI-SSD can effectively improve the accuracy of hand detection compared with the traditional SSD network. (3)In the human pose estimation sub-network, the positions of the right and left hands were estimated using pose estimation method and human body FK tree. (4)An effective result fusion method was designed. This method can effectively fuse the outputs of the two sub-networks to distinguish and locate human's two hands.

The experiments on the proposed parallel deep neural network method mainly have the following contributions: (1)The experimental results of hand detection and location showed that the proposed RI-SSD network can effectively improve the accuracy of hand detection and ensure real-time performance. (2)The results of the dual-hand position estimation experiment showed that the use of using the human FK tree can effectively estimate the positions of the left and right hands. (3)The results of the dual-hand detection and location experiment showed that the proposed parallel deep neural network and fusion method can improve the accuracy of hand detection and distinguish and locate the left and right hands effectively. (4)A set of SRSSL was created for the interaction between the astronaut and AAR, and the parallel deep neural network was applied to the astronaut-robot interaction platform. The results showed the method performs effectively in the space HRI system.

The proposed parallel deep neural network can distinguish and locate the astronaut's two hands accurately. However, there are still some areas still need improvement. The current method can only be employed to the detection and location of both hands and cannot detect the hands when they are severely occluded in the experiment. Thus, the dual-hand tracking method will be further enhanced.

References

[1] Q. Gao, J. Liu, Z. Ju, Y. Li, T. Zhang, and L. Zhang, "Static hand gesture recognition with parallel cnns for space human-robot interaction," in *International Conference on Intelligent Robotics and Applications*, pp. 462–473. Springer, 2017.

[2] Y.-W. Bai, W.-C. Hsu, and C.-C. Chan, "Design and implementation of a four-quadrant and voice interaction user interface of a smartphone for the visually impaired users," in *Consumer Electronics-Berlin (ICCE-Berlin), 2016 IEEE 6th International Conference on*, pp. 190–192. IEEE, 2016.

[3] B. J. A. Rani and A. Umamakeswari, "Electroencephalogram-based brain controlled robotic wheelchair," *Indian Journal of Science and Technology*, vol. 8, no. S9, pp. 188–197, 2015.

[4] J. L. Raheja, R. Shyam, U. Kumar, and P. B. Prasad, "Real-time robotic hand control using hand gestures," in *Machine Learning and Computing (ICMLC), 2010 Second International Conference on*, pp. 12–16. IEEE, 2010.

[5] K. R. Konda, A. Königs, H. Schulz, and D. Schulz, "Real time interaction with mobile robots using hand gestures," in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, pp. 177–178. ACM, 2012.

[6] A. Chaudhary, J. L. Raheja, K. Das, and S. Raheja, "Intelligent approaches to interact with machines using hand gesture recognition in natural way: a survey," *arXiv preprint arXiv:1303.2292*, 2013.

[7] M. Kölsch and M. Turk, "Robust hand detection." in *FGR*, pp. 614–619, 2004.

[8] F. Duan, L. Dai, W. Chang, Z. Chen, C. Zhu, and W. Li, "semg-based identification of hand motion commands using wavelet neural network combined with discrete wavelet transform." *IEEE Trans. Industrial Electronics*, vol. 63, no. 3, pp. 1923–1934, 2016.

[9] T. Ince, S. Kiranyaz, L. Eren, M. Askar, and M. Gabbouj, "Real-time motor fault detection by 1-d convolutional neural networks." *IEEE Trans. Industrial Electronics*, vol. 63, no. 11, pp. 7067–7075, 2016.

[10] S. Kiranyaz, A. Gastli, L. Ben-Brahim, N. Alemadi, and M. Gabbouj, "Real-time fault detection and identification for mmc using 1d convolutional neural networks," *IEEE Transactions on Industrial Electronics*, 2018.

[11] K. Lenc and A. Vedaldi, "R-cnn minus r," *arXiv preprint arXiv:1506.06981*, 2015.

[12] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, pp. 91–99, 2015.

[14] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[15] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*, pp. 21–37. Springer, 2016.

[16] X. Zabulis, H. Baltzakis, and A. A. Argyros, "Vision-based hand gesture recognition for human-computer interaction." *The universal access handbook*, vol. 34, p. 30, 2009.

[17] T. H. N. Le, K. G. Quach, C. Zhu, C. N. Duong, K. Luu, M. Savvides, and C. B. Center, "Robust hand detection and classification in vehicles and in the wild," in *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1203–1210, 2017.

[18] T. H. N. Le, C. Zhu, Y. Zheng, K. Luu, and M. Savvides, "Robust hand detection in vehicles," in *Pattern Recognition (ICPR), 2016 23rd International Conference on*, pp. 573–578. IEEE, 2016.

[19] G. Cao, X. Xie, W. Yang, Q. Liao, G. Shi, and J. Wu, "Feature-fused ssd: fast detection for small objects," in *Ninth International Conference on Graphic and Image Processing (ICGIP 2017)*, vol. 10615, p. 106151E. International Society for Optics and Photonics, 2018.

[20] Y. Lee, H. Kim, E. Park, X. Cui, and H. Kim, "Wide-residual-inception networks for real-time object detection," in *Intelligent Vehicles Symposium (IV), 2017 IEEE*, pp. 758–764. IEEE, 2017.

[21] M. Zhao, M. Kang, B. Tang, and M. Pecht, "Deep residual networks with dynamically weighted wavelet coefficients for fault diagnosis of planetary gearboxes," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4290–4300, 2018.

[22] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "Dssd: Deconvolutional single shot detector," *arXiv preprint arXiv:1701.06659*, 2017.

[23] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4724–4732, 2016.

[24] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," *arXiv preprint arXiv:1611.08050*, 2016.

[25] D. Tang, Q. Ye, J. Taylor, S. Yuan, P. Kohli, C. Keskin, T.-K. Kim, and J. Shotton, "Opening the black box: Hierarchical sampling optimization for hand pose estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[26] A. Mittal, A. Zisserman, and P. H. Torr, "Hand detection using multiple proposals." in *BMVC*, pp. 1–11. Citeseer, 2011.

[27] S. Bambach, S. Lee, D. J. Crandall, and C. Yu, "Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1949–1957, 2015.

[28] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1532–1545, 2014.

[29] J. Liu, Q. Gao, Z. Liu, and Y. Li, "Attitude control for astronaut assisted robot in the space station," *International Journal of Control, Automation and Systems*, vol. 14, no. 4, pp. 1082–1095, 2016.

[30] Q. Gao, J. Liu, T. Tian, and Y. Li, "Free-flying dynamics and control of an astronaut assistant robot based on fuzzy sliding mode algorithm," *Acta Astronautica*, vol. 138, pp. 462–474, 2017.

[31] J. Liu, Y. Luo, and Z. Ju, "An interactive astronaut-robot system with gesture control," *Computational intelligence and neuroscience*, vol. 2016, 2016.