A Method for Detecting Interaction between 3D Hands and Unknown Objects in RGB Video

Yankun Wu\textsuperscript{1,3}, Peipei Wang\textsuperscript{2*}, Xin Wang\textsuperscript{3}, Guanqun Liu\textsuperscript{1,3}, Lin Zhao\textsuperscript{3} and Ji Xiang\textsuperscript{3}

\textsuperscript{1}School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100049, China
\textsuperscript{2}National Internet Emergency Center, Beijing 100029, China
\textsuperscript{3}Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
\textsuperscript{*}Corresponding author’s e-mail: wangpeipei@cert.org.cn

Abstract. We propose a model that can extract 3D position of hand and object in per-frame of RGB videos through a single feed-forward neural network and a zero-shot learning classifier, and understand unknown hand-object interactions in the entire video through an interactive temporal module. The process is trained end-to-end, without depth images or annotated coordinates as input, which has good application prospects in real life.

1. Introduction
Nowadays, most research on hand recognition focuses on recognizing the posture and the shape of the hand without interacting with an object. However, most hand movements in life will interact with the object, in this way, the mutual occlusion of the hand and the object increases the difficulty of hand detection. Also, the semantic information of hand action is of great significance to our understanding of human actions. In the meantime, the types of objects interacting with hand in the existing hand dataset are very limited. So it is very realistic to propose a method that can identify the interaction between hands and unknown objects based on RGB video.

This paper proposes a model that takes RGB videos as input and outputs the 3D posture of hands and objects, hand action categories, and object categories, without depth images or real coordinates of objects as input, which has the potential to be widely used in real life.

2. Related Work
2.1. Hand Detection
The traditional method of recognizing hand is mainly to separate the hands from the first perspective to recognize their gestures [1]; or recognize the hand gestures in the RGB images of the first and third perspectives[2], but these methods do not model the interaction between hand and object. Some methods use object interaction as an additional constraint when estimating hand motions [3], which improves the accuracy of hand motion recognition, but relies on depth images as input, otherwise the accuracy is very low; some methods reconstruct the hand posture[4], the edges of the object are well restored, but the semantic information is not learned; some methods can recognize the interaction...
between the hand and the object[5], but can only recognize the known objects in the dataset therefore lack generalization.

2.2. Visual relationship detection

The Visual relationship detection not only identifies the objects and their positions in the image, but also the relationship between the objects. [6] proposed relational reasoning network surpassed humans in reasoning about object relations in the visual question answering task set, showing the great potential of neural networks in reasoning relational tasks.

In the method of recognizing the interaction between people and objects, [7] uses an object detection model to first recognize people and objects, and then for each pair of <people, object>, the visual likelihood and language of the visual model are calculated separately. The semantic likelihood of the model is used to predict the predicate verbs in it. In order to make the model have the transfer ability, such as the migration of "human-riding-horse" to "human-riding-elephant", zero-shot learning module is added, which projects the word vectors corresponding to the two objects in the object pair into a k-dimensional vector to represent the semantic likelihood of the relationship between the two objects, so that the model predicts the relationship more accurately.

3. Model

Our model transfers the method of human-object relationship detection to hand-object relationship recognition, and introduces a zero-shot learning classifier module to identify unknown object categories that interact with hands. The entire model contains three modules: action and object detection module, zero-shot learning classifier module and interactive temporal module. The model architecture is shown in figure 1.

![Figure 1 The model architecture](image)

3.1. Action and object detection module

We have specified 21 key points for the hand and the object. The key points of the hand are the four joints of each finger and the wrist node, while the key points of the object are taken from the boundary of the eight vertices, as well as the center point and the midpoint of the 12 sides of the frame. We divide each picture frame into a H × W grid, and extend D grids to the depth (H, W, D represent height, width, and depth respectively), using pixels as the unit on the plane and meters as the unit in the depth direction. In this grid coordinate system, we use the upper left corner of the grid as the origin of the
coordinate system, and each grid is a unit. The cell coordinates are the number of cells offset from
the upper left corner.

In order to jointly predict the posture and category of the hand and the object at the same time, in
the grid coordinate system, the position of the hand and the object and the vector diagram stored in the
cell where it is located. We store two vectors in each cell to predict the characteristics of the hand and
the object respectively, \( \mathbf{v}_i^h \) and \( \mathbf{v}_i^o \) are the coordinates of the key points of the hand and the object
respectively, \( \mathbf{v}_i^h = (y_i^h, p_i^h, c_i^h) \), \( \mathbf{v}_i^o = (y_i^o, p_i^o, c_i^o) \), and \( N_c \) is the number of the key points of
the hand or the object. \( \mathbf{p}_a^a \) is the action category probability, \( N_a \) is the number of action category, \( \mathbf{P}_i^o \)
is the object category probability, and \( N_o \) is the number of object category. The grid where the wrist
node and the center point of the object are located is used to predict the type of action and object. Also,
we add a background category, if the object is unknown objects, it will be divided into background
categories, and then enter the zero-shot learning classifier to identify its category. \( c_i^h, c_i^o \) are the
confidence values. The two vectors stored in each cell are obtained by a feed-forward neural network.
We first determine the coordinates of the cell \((u_i, v, z)\) where the wrist node and the center of object are
located, and then predict the other key points’ offset \( \Delta u, \Delta v, \Delta z \) of the key point in three
dimensions relative to the upper left corner of the key point cell, then the coordinates \( \hat{w}_i \) of the key
point in the grid coordinate system can be calculated:

\[
\hat{w}_i = g(\Delta i) + i \quad (i = u, v, z)
\]

Among them, since the cells where the wrist node and the center point of the object are located are
responsible for predicting the action and the object category, we use \( g(x) \) to control the offset of these
two points between \([0,1]\). The expression of \( g(x) \) is as follows:

\[
g(x) = \begin{cases} 
\text{sigmoid}(x), & \text{if } x \in \text{key control points} \\
0, & \text{otherwise}
\end{cases}
\]

\( g(x) \) represents the function that constrains the offset between the wrist node and the center point of
the object, \( x \) represents the offset of the key point relative to the upper left corner of the cell in three
dimensions, sigmoid represents the activation function, it can map a real number to the interval of \([0,1]\). We use this function to make the wrist node and the center point of the object fixed in the cell to
predict the action and object category. In addition, with the three-dimensional position in the grid
coordinate system and the camera internal parameter \( K \), we can calculate the three-dimensional
coordinates of the key point in the camera coordinate system as:

\[
\hat{y} = \hat{w}_z \cdot C_z \cdot K^{-1} [\hat{w}_u \cdot C_u, \hat{w}_v \cdot C_v, 1]^T
\]

We set the confidence function as:

\[
c(x) = \begin{cases} 
\left(\frac{e^{\alpha(1-D_T(x)d_{th}^{-1})} - 1}{e^\alpha - 1}\right)^{-1}, & D_T(x) < d_{th} \\
0, & D_T(x) \geq d_{th}
\end{cases}
\]

\( D_T(x) \) is the Euclidean distance between the predicted point and the real point, \( \alpha \) represents the
hyper parameter, \( d_{th} \) represents the set threshold. When the predicted value is closest to the true value,
the greater \( c(x) \) is, and the greater confidence will be. The total confidence is:

\[
c = 0.5 \cdot c_{uv} (\hat{w}_u, \hat{w}_v) + 0.5 \cdot c_z (\hat{w}_z)
\]

The total loss function of action and object detection module is:

\[
L = \lambda_{\text{pose}} \sum_{i \in G^t} \left( \|\hat{y}_i^h - y_i^h\|^2 + \|\hat{y}_i^o - y_i^o\|^2 \right) + \lambda_{\text{conf}} \sum_{i \in G^t} \left( (c_i^h - \hat{c}_i^h)^2 + (c_i^o - \hat{c}_i^o)^2 \right)
- \lambda_{\text{act}} \sum_{i \in G^t} \hat{p}_i^a \log \hat{p}_i^a - \lambda_{\text{obj}} \sum_{i \in G^t} \hat{p}_i^o \log \hat{p}_i^o
\]
\( \lambda_{\text{pose}} \) represents the loss function parameter for predicting the position of the hand and the object, \( \lambda_{\text{conf}} \) represents the loss function parameter for the confidence, \( \lambda_{\text{actcls}} \) represents the loss function parameter for predicting the action category, \( \lambda_{\text{objcls}} \) represents the loss function parameter for predicting the object category, and \( G_i \) represents the grid for dividing pictures. \( \hat{y}_i^h \) represents the predicted hand coordinates, \( \hat{y}_i^o \) represents the predicted object coordinates, \( \hat{c}_i^h \) represents the confidence of the predicted hand movement category, \( \hat{c}_i^o \) represents the confidence of the predicted object category, \( \hat{p}_i^h \) represents the predicted object category probability, and \( \hat{p}_i^a \) represents the predicted action category probability.

3.2. Zero-shot learning classifier module

The zero-shot learning module is used to identify unknown object categories in the testing phase. After detecting the 6D position of the object in the image, then we score the detected object for each category. If the background score is the highest, the object is considered to be an unknown category. Subsequently, we use its semantic information to find the closest object category in the semantic space to predict its category. We multiply the scores of other prediction classes except the background with their vectors in the semantic space, and then add up these semantic vectors to form the final semantic vector of the unknown object. In this way, we can calculate the similarity of the categories in the semantic space. When the highest similarity is not lower than the threshold, the unknown object is considered to belong to the class with the highest similarity.

3.3. Interactive temporal module

Since the action and object detection module only learns the information of each frame of the image, and does not use the temporal information in the video, we add an interactive recurrent neural network module part. Firstly, we input a multilayer perceptron to model their relationship, and then use it as the input of the interactive recurrent network. The model of the recurrent network is as follows:

\[
f_\phi(g_\theta(y_i^h, y_i^o))
\]

\( f_\phi \) is a recurrent neural network model, while \( g_\theta \) is a multi-layer perceptron model, and finally the output is the interaction category of hand and object in this video.

4. Experiment

The experimental phase of this model is divided into training phase and testing phase.

In the training process, we train the action and object detection module and interactive temporal module in two steps. First, the video frames are used as input to train the action and object detection module, which is based on a feed-forward neural network. The output is the key point coordinates of hands and objects, hand action category, and object category for each image. After training the action and object detection module, we fix its parameter and then train the interactive temporal module. We input the key point vectors of the hands and objects obtained through a multi-layer perceptron to learn their interaction relationship, and then through a recurrent neural network, and finally output the interactive category estimation in the video.

In the testing process stage, the complete model takes a series of video frames as input. Each frame of image is first passed through the action and object detection module, and the key point vectors of the hands and objects, the category of hand and object can be obtained. If the predicted object category is the background category, the category of object will be predicted through the zero-shot learning classifier. And then the key points are modeled by a multilayer perceptron to get the relation vector, and then pass the obtained vector through a two-layer hidden layer recurrent neural network to learn the temporal information in the video, and finally output the interactive category estimate. Therefore, the 3D posture of the hand and the object in each frame and the interaction category of the hand and the object in the entire sequence can be estimated.
For the entire model, we first train action and object detection module and interactive temporal module. After training the parameters and fixing them, we can use the trained complete model, input a RGB video, and output the 3D position of the hand and the object in each frame of the video and the interaction between the hand and the object in the entire video.

We divide the training set into the train set and the test set according to the types of objects interacting with the hand, where the test set includes object categories (unknown classes) that do not appear in the training set.

5. Conclusion
This paper proposes a model which can extract the 3D position of hand and unknown object in each frame of the RGB video, and identify the category of hand-object interaction, thereby understanding the semantic information of the entire video without depth image or real coordinate data as input. It solves the problem that the semantic information interacting with the object cannot be recognized in traditional gesture recognition, which provides a good theoretical basis for its wide application. Besides, the model detects the position trajectory of hand and object, and the estimation of the action category and the object category, which can be applied to anomaly action detection. Moreover, this model can detect unknown object categories that are not in the dataset, making a wider application range in life. Since there is no suitable dataset, in the future, we will create a dataset for experiments and then apply this model.

Acknowledgments
This work is partially supported by National Natural Science Foundation of China (No. U1636220).

References
[1] GARCIA-HERNANDO G, YUAN S, BAEK S, et al. First-person hand action benchmark with rgb-d videos and 3d hand pose annotations[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 409-419.
[2] Iqbal U, Molchanov P, Gall T B J, et al. Hand pose estimation via latent 2.5 d heatmap regression[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018: 118-134.
[3] CHOI C, HO YOON S, CHEN C N, et al. Robust hand pose estimation during the interaction with an unknown object[C]//Proceedings of the IEEE International Conference on Computer Vision. 2017: 3123-3132.
[4] HASSON Y, VAROL G, TZIONAS D, et al. Learning joint reconstruction of hands and manipulated objects[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 11807-11816.
[5] TEKIN B, BOGO F, POLLEFEYS M. H+O: Unified egocentric recognition of 3d hand-object poses and interactions[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 4511-4520.
[6] A. SANTORO, D. RAPOSO, D. G. BARRETT, M. MALINOWSKI, R. PASCANU, P. BATTAGLIA, and T. LILLICRAPH. A Simple Neural Network Model for Relational Reasoning. In NIPS, 2017.
[7] LU C, KRISHNA R, BERNSTEIN M, et al. Visual relationship detection with language priors[C]//European conference on computer vision. Springer, Cham, 2016: 852-869.