IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 5, 2021, accepted February 16, 2021, date of publication February 22, 2021, date of current version March 4, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3061208

Improved Human-Object Interaction Detection Through On-the-Fly Stacked Generalization

GEONU LEE[1], KIMIN YUN[2], AND JUNGCHAN CHO[1], (Member, IEEE)

[1]College of Information Technology, Gachon University, Seongnam 13120, South Korea
[2]Artificial Intelligence Research Laboratory, Electronics and Telecommunications Research Institute, Daejeon 34129, South Korea

Corresponding author: Jungchan Cho (thinkai@gachon.ac.kr)

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korean Government (MSIT) under Grant NRF-2019R1F1A1058666, and in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by the Korean Government (MSIT), Development of a High-Performance Visual Big Data Discovery Platform for Large-Scale Real-Time Data Analysis under Grant B0101-15-0266.

ABSTRACT Human-object interaction (HOI) detection, which finds the relationships between humans and objects, is an important research area, but current HOI detection performance is unsatisfactory. One of the main problems is that CNN-based HOI detection algorithms fail to predict correct outputs for unseen test data based on a limited number of available training examples. Herein, we propose a novel framework for HOI detection called the on-the-fly stacked generalization deep neural network (OSGNet). OSGNet consists of three main components: (1) feature extraction modules, (2) HOI relationship detection networks, and (3) a meta-learner for combining the outputs of sub-models. Here, components (1) and (2) are considered to be sub-models. Any task-based feature extraction modules, such as classification or human pose estimation modules, can be used as sub-models. To achieve on-the-fly stacked generalization, the sub-models and meta-learner are trained simultaneously. The sub-models are trained to provide complementary information, and the meta-learner improves the generalization performance for unseen test data. Extensive experiments demonstrate that the proposed method achieves state-of-the-art accuracy, particularly in cases involving rare classes.

INDEX TERMS Deep learning, human-object interaction, human pose estimation, action recognition.

I. INTRODUCTION

Object detection technology based on deep learning has developed very rapidly. In recent years, significant research has been conducted to develop a comprehensive understanding of various types of scenes. In particular, human-object interaction (HOI) detection can further our understanding of various scenes. HOI detection can provide important information for many applications, including human-robot interaction, autonomous vehicles, and abnormal behavior recognition [1]–[3], [32]–[34].

However, the HOI detection problem is difficult to solve for the following reasons:

- It can occur in a large number of different situations compared to the number of possible combinations of people and objects.
- Assuming that there are one person and one object in an image, there can be multiple interactions between them, such as *riding* a bicycle and *sitting* on the bicycle. In other words, HOI detection must detect multiple objects and recognize multiple labels according to their combinations simultaneously.
- It is difficult to ensure that model training data contain correct labels for all possible combinations.

For HOI detection, many methods [5], [7], [13], [15], [17], [24], [29], [30], [35] have been proposed to overcome these challenges, but HOI detection performance is still unsatisfactory and remains an open problem. One of the major issues is that convolutional neural network (CNN)-based HOI detection algorithms fail to produce correct outputs for unseen test data based on a limited number of available training samples, particularly for rare classes, such as ''point'' and ''wash a knife''. Therefore, it is desirable to design a CNN structure to improve generalization accuracy for HOI detection.

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaohui Yuan.

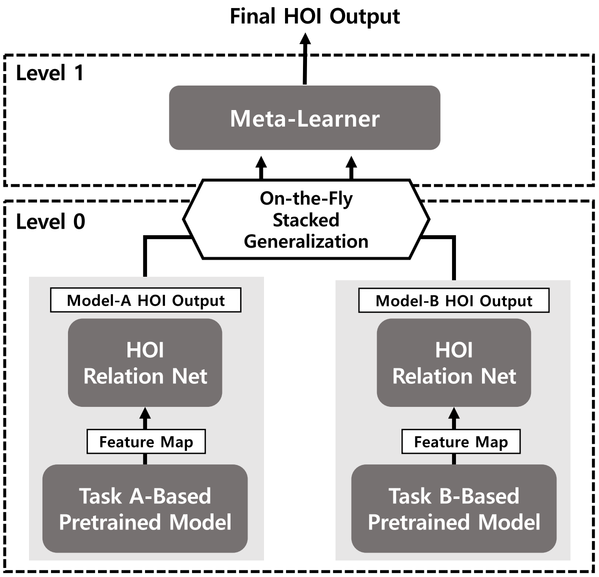


FIGURE 1. **Overview: Stacking ensemble. Unlike a general stacking ensemble, where levels 0 and 1 are trained independently, the proposed scheme simultaneously trains the sub-models at level 0 and meta-learner at level 1, allowing for an "on-the-fly" stacking ensemble.**

In this article, we propose a novel framework for HOI detection called the on-the-fly stacked generalization deep neural network (OSGNet). A stacked generalization is a type of ensemble technique that is a well-known approach to improving generalization capabilities in machine learning [31]. Such an ensemble consists of two stacked levels called level 0 and level 1, as shown in Figure 1. While "bagging" and "boosting" approaches can only utilize one type of algorithm at a time, stacked generalization improves generalization accuracy by combining different types of algorithms (level 0) using a meta-learner (level 1) [20]. To the best of our knowledge, there has been no prior work analyzing stacked generalization for HOI detection.

The main advantage of stacked generalization is that it attempts to learn how to combine predictions optimally to compensate for each sub-model's weaknesses by using the outputs of sub-models as inputs. In this type of ensemble, the features from different sub-models should contain different information. For HOI detection, analyzing human body parts can aid significantly in understanding the relationship between a person and an object. This is because when a person interacts with an object, the interaction mainly occurs close to body parts such as the head, hands, and feet. Based on this intuition, we propose a method for using different feature extraction models trained for different tasks of classification (*e.g.*, ResNet [11]) and human pose estimation (*e.g.*, HRNet [28]).

An overview of the OSGNet architecture for HOI detection is presented in Figure 1. The OSGNet consists of three main components: (1) feature extraction modules, (2) HOI relationship detection networks, and (3) a meta-learner for combining the outputs of sub-models. Here, components (1) and (2) are considered to be sub-models and any HOI network can be used as a sub-model. In this study, sub-model A is set to a VSGNet [29] that learns HOIs using a graph convolutional network. For sub-model B, we propose a human-pose-based HOI network inspired by VSGNet [29]. Additionally, unlike conventional stacked generalization, which independently trains sub-models at level 0 and a meta-learner at level 1, the proposed scheme simultaneously trains the sub-models and meta-learner, thereby facilitating on-the-fly stacked generalization. This allows sub-models to be trained such that complementary information can be passed into the inputs of the meta-learner.

Extensive experiments were performed to compare the proposed method to a baseline method (*i.e.*, VSGNet [29], which is sub-model A without stacked generalization) on two popular benchmark datasets: V-COCO [9] and HICO-DET [4]. The results demonstrate that the proposed OSGNet achieves state-of-the-art performance, particularly for cases involving rare classes.

Our main contributions can be summarized as follows:

- Most HOI detection methods combine image and human pose information using a single underlying model and do not leverage the advantages of using multiple diverse models. Unlike such approaches, we propose a stacked-generalization-based framework for combining image and human pose information, which results in higher generalization accuracy.
- The proposed scheme simultaneously trains sub-models and a meta-learner, which allows the sub-models to be trained such that complementary information can be passed into the inputs of the meta-learner.
- Extensive experiments validate the proposed method's achievement of state-of-the-art accuracy, particularly for cases involving rare classes.

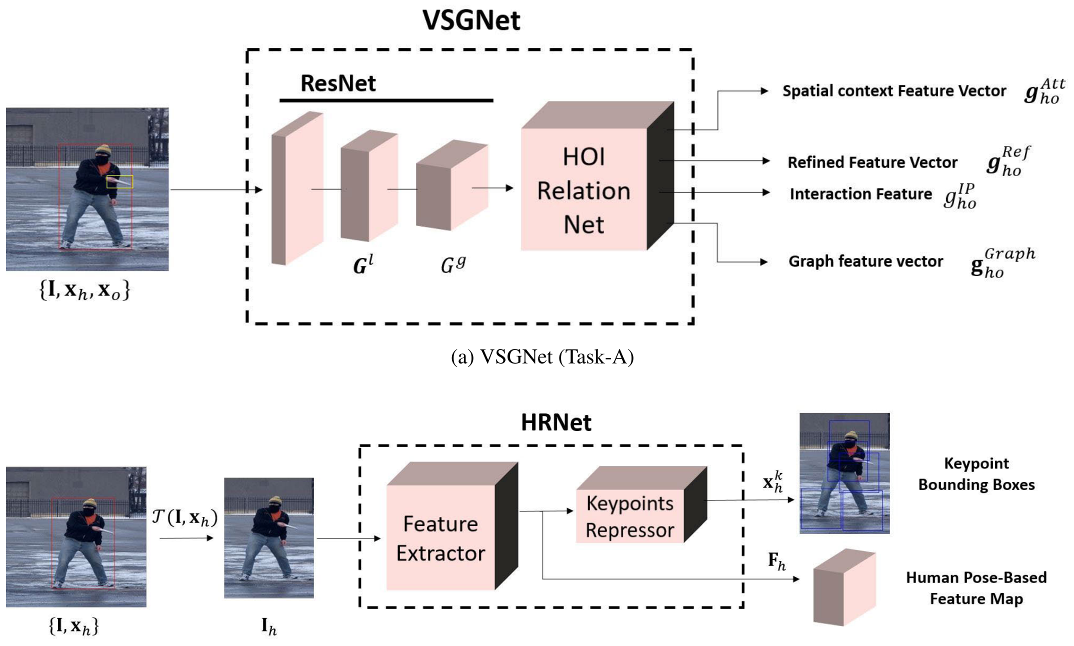
The remainder of this article is organized as follows. Section II introduces related works on HOI methods. Section III describes the proposed OSGNet. Section IV presents implementation details and experimental results. Finally, Section V concludes this article.

II. RELATED WORK

A. HUMAN-OBJECT INTERACTION DETECTION

HOI detection addresses the task of detecting <human, verb, object> triplets within a given image. In most cases, the appearance of a person or object contains useful clues regarding the parts of an image that are relevant to interaction prediction. Gkioxari *et al.* [7] proposed a human-centered approach called InteractNet, which was designed based on the hypothesis that a person's appearance is a powerful signal for the localization of the object they are interacting with. Gao *et al.* [5] proposed an instance-centric attention module that selectively aggregates features that are relevant to detecting HOIs.

Context modeling of the spatial configurations between humans and objects is also useful for HOI analysis.


VSGNet
ResNet
HOI Relation Net
G^l
G^g
$\{\mathbf{I}, \mathbf{x}_h, \mathbf{x}_o\}$
Spatial context Feature Vector g_{ho}^{Att}
Refined Feature Vector g_{ho}^{Ref}
Interaction Feature g_{ho}^{IP}
Graph feature vector $\mathbf{g}_{ho}^{Graph}$
HRNet
$\{\mathbf{I}, \mathbf{x}_h\}$
$\mathcal{T}(\mathbf{I}, \mathbf{x}_h)$
$\mathbf{I}_h$
Feature Extractor
Keypoints Repressor
$\mathbf{x}_h^k$
$\mathbf{F}_h$
Keypoint Bounding Boxes
Human Pose-Based Feature Map

(a) VSGNet (Task-A)

(b) HRNet

FIGURE 2. (a) VSGNet [29], where the feature extractor is a residual network. (b) HRNet [28].

Qi *et al.* [24] incorporated structural knowledge using a graph-parsing neural network (GPNN). Chao *et al.* [4] proposed interaction patterns that characterize the spatial relationships between human and object bounding boxes as binary images with two channels. A similar concept was introduced as the "box attention mechanism" in [15]. Ulutan *et al.* [29] proposed VSGNet architecture, which models the relative spatial and structural connections between objects. As mentioned previously, our proposed method uses the VSGNet architecture as sub-model A.

Recently, significant research has been conducted to improve HOI detection speed and accuracy. Liao *et al.* [18] proposed a single-stage HOI detection method by parallelizing HOI detection and matching. However, the performance comparisons in Section IV demonstrate that a single-stage-based method [18] achieves significantly weaker performance compared to the proposed method for rare data classes.

B. POSE INFORMATION FOR DETECTING HOI

Human poses are used in many studies because they are important cues for the analysis of HOI. The multi-level relation detection strategy proposed by Wan *et al.* [30] utilizes human poses to capture the global spatial compositions of relationships, which are used in an attention mechanism at the level of human body parts. Zhou and Chi [35] proposed a novel relational parsing neural network (RPNN) represented by an object-body part graph and a human-body part graph. The RPNN model can implicitly analyze pairwise relationships in two graphs in an unsupervised manner. Gupta *et al.* [10] demonstrated that a simple factorized model can outperform more sophisticated models for HOI detection. Recent attempts have been made to improve HOI performance by leveraging additional information. Li *et al.* [16] used 3D human poses instead of 2D human poses. Li *et al.* [37] proposed an integration-decomposition network that analyzes HOI semantics in the transformation function space. These methods could be used as a sub-model for the stacked generalization.

Although progress has been made in terms of improving HOI detection in a fully supervised manner, it is impractical to label training data for all possible interactions between humans and objects. Therefore, various studies have been conducted to overcome information limitations in HOI datasets. In these types of studies, detailed cues such as human poses have been considered. Li *et al.* [17] explored interactive knowledge indicating whether humans and objects interact with each other. Their conclusion was that regardless of the setting of an HOI category, interaction knowledge can be learned across HOI datasets. Shen *et al.* [27] extended HOI awareness to the long tail of HOI categories using zero-shot learning based on verb-object factorization. Kim *et al.* [13] focused on correlations as action co-occurrences in images to achieve more effective training, particularly for rare classes.

We have developed a stacked generalization scheme that can be used to overcome the performance degradation resulting from a lack of data.

III. APPROACH

The proposed stacked generalization scheme can be used with any feature maps and sub-models, but for the convenience of explanation, we will assume the following. 1) Sub-model A

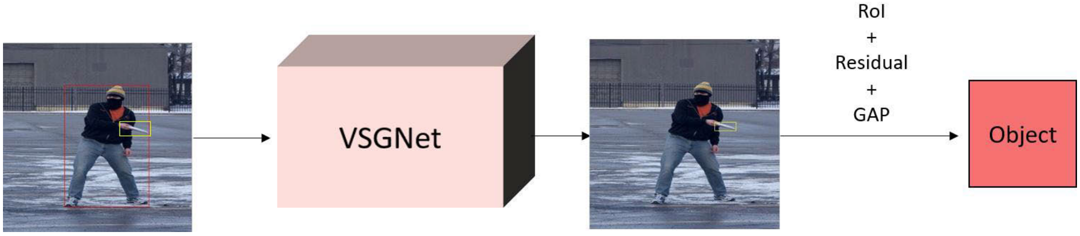


FIGURE 3. Example graphical representation of a sequence for region of interest (ROI) pooling, a residual block, and global average pooling (GAP).

in Figure 1 is based on the feature map of a residual network (*ResNet*) [11] pre-trained for a classification task and 2) sub-model *B* is based on the feature map of an *HRNet* [28] pre-trained for a human pose estimation task.

A. NOTATIONS

Let $\mathbf{I}$ represent an input image, as depicted in Figure 2a. The feature map $\mathbf{G}^g$ represents the inputs, where g represents global features for the HOI relation network in sub-model *A*, as depicted in Figure 1. This feature map is obtained from the classification task network (*i.e.*, $\mathbf{G}^g = ResNet(\mathbf{I})$). The bounding boxes for humans $h \in [1, H]$ and objects $o \in [1, O]$ are represented by $\mathbf{x}_h$ and $\mathbf{x}_o$, respectively, where H and O denote the numbers of detected humans and objects in an input image $\mathbf{I}$, respectively. The bounding boxes $\mathbf{x}_h$ and $\mathbf{x}_o$ can be obtained using any off-the-shelf object detector [26].

We assume that a single-person human pose estimator is used for sub-model *B*. A human-centric image $\mathbf{I}_h$ is required for human h. This image is obtained through an affine transformation $\mathcal{T}(\cdot)$ of the input image $\mathbf{I}$ with a human bounding box $\mathbf{x}_h$ (*i.e.*, $\mathbf{I}_h = \mathcal{T}(\mathbf{I}, \mathbf{x}_h)$), as depicted in Figure 2b. The human pose estimator provides a human-pose-based feature map $\mathbf{F}_h$ for a human h, as well as bounding boxes $\mathbf{x}_h^k$ of keypoints $k \in [1, K]$ on the human h (*i.e.*, $\{\mathbf{F}_h, \mathbf{x}_h^k\} = HRNet(\mathbf{I}_h)$), as shown in Figure 2b. K represents the number of keypoints on a human (*e.g.*, five keypoints corresponding to the head, left arm, right arm, left ankle, and right ankle). Another classification-based feature map $\mathbf{G}^l$, where "l" indicates "local features" in Figure 2a, is also used for sub-model *B*.

Finally, Figure 3 presents an example graphical representation of the generation of a feature vector for an object. The pink cube represents a VSGNet [29] and the second arrow pointing to an image with a yellow box indicates that an object feature map is pooled within the object bounding box (*i.e.*, a region of interest (ROI) pooling [6]). Next, a residual block [11] and a global average pooling (GAP) [8] operation are used to generate an object feature vector. We also use this type of graphical representation to explain the proposed sub-model *B*, as shown in Figure 4. The other notation in Section III follows that in Table 1 if there are no additional clarifications.

B. OVERVIEW

The sub-model *A* (VSGNet) consists of a feature map extractor and HOI relation network, as shown in Figure 2a. The HOI relation network consists of visual, spatial, and graph convolutional branches, and returns three types of feature vectors. The first type of feature vector, denoted as $\mathbf{g}_{ho}^{Ref}$, is a visual feature vector refined by spatial information. The second type of feature vector, denoted as $\mathbf{g}_{ho}^{Att}$, is a spatial attention feature vector based on the coordinates of human and object locations. The third type of feature vector, denoted as $\mathbf{g}_{ho}^{Graph}$, is a relational feature vector between humans and objects [29]. The VSGNet also returns another feature for calculating the interaction probability between a human and object (for additional details, please see [29]). These four features are used as the inputs for a meta-learner, as described in Section III-D.

Sub-model *B* is designed with the same architecture as sub-model *A* (VSGNet), as shown in Figure 4. Sub-model *B* also consists of a feature map extractor and HOI relation network. The difference compared to sub-model *A* is that the visual and spatial feature vectors are based on the human-pose-based feature map $\mathbf{F}_h$ and keypoint bounding boxes $\mathbf{x}_h^k$ of the human body. These vectors are obtained from the HRNet, as shown in Figure 2b. We discuss the HOI relation network in sub-model *B* in detail in Section III-C based on the explanatory method proposed in [29] to avoid confusion.

C. LEVEL 0: SUB-MODEL B: HRNet-BASED VSGNet

Because sub-model *B* is a variant of a VSGNet with human pose information, it also has three branches for extracting visual, spatial, and graph-based relationship features of humans and objects.

1) VISUAL BRANCH

The main role of this branch is to extract visual features from the inputs of human pose and object pairs. Visual features consist of human-pose- and keypoint-based features, an object feature, and a context feature representing an entire image. For the object feature $\mathbf{g}_o$ and context feature $\mathbf{g}_C$, we reuse features from sub-model *A*. These features are extracted from a classification-based pre-trained model as follows.

Given an object bounding box $\mathbf{x}_o$, the features of the corresponding object region are extracted using *ROI* pooling. A residual block [11] and *GAP* operations are applied to extract a visual feature vector $\mathbf{g}_o$ for an object of size *R*.

$$\mathbf{g}_o = GAP(Res_O(ROI(\mathbf{G}^g, \mathbf{x}_o))), \tag{1}$$

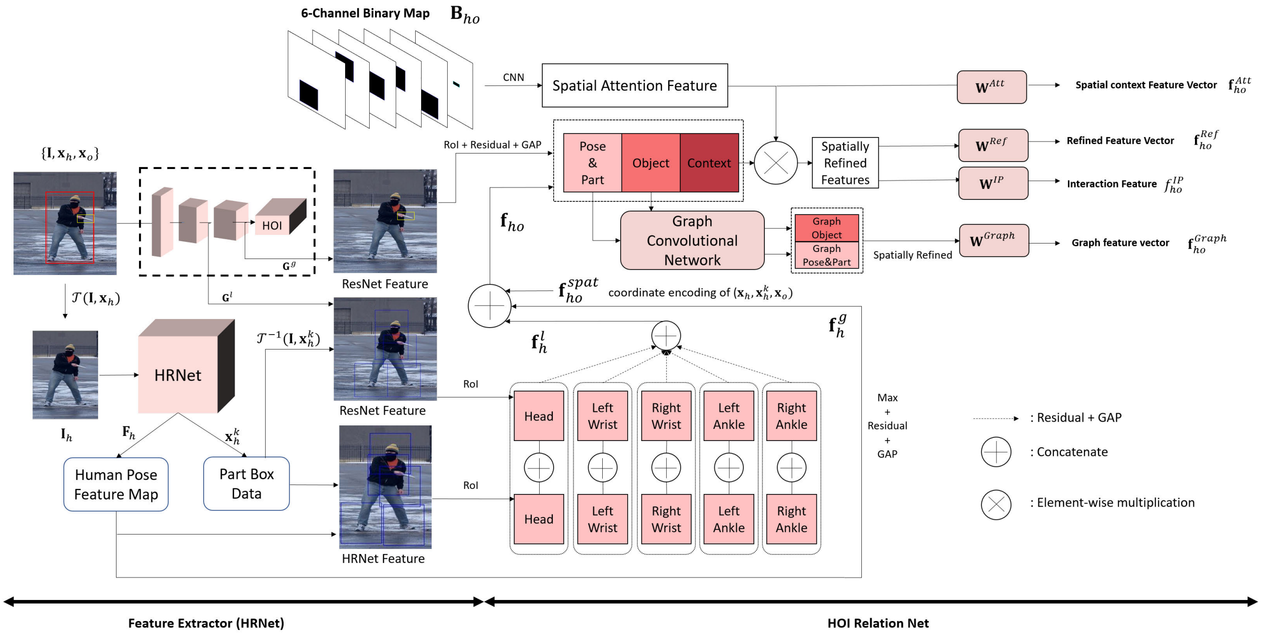


FIGURE 4. Graphical representation explaining the proposed sub-model *B*.

TABLE 1. Symbols used in this article.

Symbol	Description	Symbol	Description
$\mathbf{G}^{\{\}}_{\{\}}$	feature map from the classification network	$\mathbf{F}^{\{\}}_{\{\}}$	feature map from the human pose estimation network
$\mathbf{g}^{\{\}}_{\{\}}$	feature vector from the classification network	$\mathbf{f}^{\{\}}_{\{\}}$	feature vector from the human pose estimation network
$\mathbf{p}^{\{\}}_{\{\}}$	class probability vector	$\mathbf{x}_h$	bounding box for a human h
$\mathbf{W}^{\{\}}$	projection matrix (fully connected layer)	$\mathbf{x}_o$	bounding box for an object o
$Res^{\{\}}_{\{\}}$	residual block [11]	$\mathbf{x}_h^k$	k-th bounding box of keypoints for a human h)
$\mathcal{T}$	affine transformation	$\mathcal{T}^{-1}$	inverse affine transformation of $\mathcal{T}$
$ResNet$	ResNet [11]	$HRNet$	HRNet [28]
GAP	global average pooling	ROI	ROI pooling

* The {} in a superscript indicates where an item is extracted (*e.g.*, "*Att*" denotes the attention branch).
* The "ho" in the subscript {} indicates a pair consisting of a human h and object o.

where Res_O represents a residual block for an object input. A context feature vector $\mathbf{g}_C$ of size R is extracted from an entire image. The residual block and *GAP* are then applied as

$$\mathbf{g}_C = GAP(Res_C(\mathbf{G}^g)), \tag{2}$$

where Res_C represents a residual block for a context input. These two feature vectors are the same as those in sub-model *A* (*i.e.*, VSGNet [29]).

However, the visual feature vector for a human in the proposed OSGNet is designed as the following:

1) A global human pose feature vector $\mathbf{f}_h^g$ of size P is obtained from $\mathbf{F}_h$ instead of $\mathbf{G}^g$ as

$$\mathbf{f}_h^g = GAP(Res_H^g(\mathbf{F}_h)), \tag{3}$$

where Res_H^g represents the residual block for the human input.

2) Local keypoint feature maps $\mathbf{F}_h^k$ are obtained from $\mathbf{F}_h$, $\mathbf{G}^l$, and the bounding boxes $\mathbf{x}_h^k$.

$$\mathbf{F}_h^k = ROI(\mathbf{F}_h, \mathbf{x}_h^k) \oplus ROI(\mathbf{G}^l, \mathcal{T}^{-1}(\mathbf{I}, \mathbf{x}_h^k)), \tag{4}$$

where $\oplus$ is a concatenation operation and $\mathcal{T}^{-1}(\cdot)$ is the inverse affine transformation for generating keypoint bounding boxes corresponding to a feature map in sub-model *A*, as depicted in Figure 4.

3) By concatenating local keypoint feature vectors $\mathbf{F}_h^k$ of size Q, a local human pose feature vector for each human h is obtained as

$$\mathbf{f}_h^l = \oplus_{k=1}^K GAP(Res_H^k(\mathbf{F}_h^k)), \tag{5}$$

where Res_H^k represents the k-th residual block for the k-th keypoint of a human body and $\oplus_{k=1}^K$ concatenates the K feature vectors obtained by K GAP operations.

4) The visual human pose feature vector for a human h is generated through the concatenation of global and local human pose feature vectors and spatial information.

$$\mathbf{f}_{ho} = \mathbf{f}_h^g \oplus \mathbf{f}_h^l \oplus \mathbf{f}_{ho}^{spat}, \tag{6}$$

where $\mathbf{f}_{ho}^{spat}$ of size S is an encoding of $\mathbf{x}_o$ in a coordinate space relative to $\mathbf{x}_h$ and $\mathbf{x}_h^k$ [7].

Finally, all types of visual feature vectors are concatenated and projected onto a D-dimensional feature space.

$$\mathbf{f}_{ho}^{Vis} = \mathcal{W}^{Vis}(\mathbf{f}_{ho} \oplus \mathbf{g}_o \oplus \mathbf{g}_C), \tag{7}$$

where $\mathcal{W}^{Vis}$ represents two sequences of a fully connected layer and rectified linear unit ($ReLU$) layer [21], and $\mathbf{f}_{ho}^{Vis}$ is a combined visual feature vector based on the human pose estimation task, where ho represents a human h and object o pair.

2) SPATIAL ATTENTION BRANCH

This branch learns spatial interaction patterns between human keypoints and objects. Given a set of human keypoint bounding boxes $\mathbf{x}_h^k$ and an object bounding box $\mathbf{x}_o$, we generate a six-channel binary spatial configuration map $\mathbf{B}_{ho}$, as depicted in Figure 4. It should be noted that VSGNet [29] generates two binary maps based on two bounding boxes: those for a human and an object. The binary maps contain zeroes except for in the locations of human keypoints and object box coordinates $\mathbf{x}_h^k$ and $\mathbf{x}_o$, respectively. This operation is followed by six convolutional layers $Conv^{Spat}$ for analyzing the spatial configuration map of the human and object pair.

$$\mathbf{a}_{ho} = \mathbf{W}^{Spat}(GAP(Conv^{Spat}(\mathbf{B}_{ho}))), \tag{8}$$

where $\mathbf{W}^{Spat}$ is a fully connected layer for generating an attention feature vector of size D. Because the locations of an object and human keypoints are in different channels, a model using the spatial configuration map $\mathbf{B}_{ho}$ can learn the possible spatial relationships between human keypoints and an object.

The attention feature vector $\mathbf{a}_{ho}$ can be used to classify HOIs [15], [29] because it encodes spatial configurations. To this end, we generate a spatial context feature vector $\mathbf{f}_{ho}^{Att}$ of size M as

$$\mathbf{f}_{ho}^{Att} = \mathbf{W}^{Att}(\mathbf{a}_{ho}), \tag{9}$$

where $\mathbf{W}^{Att}$ is a fully connected layer. This layer is followed by a sigmoid function $\sigma(\cdot)$ for generating an action class probability $\mathbf{p}_{ho}^{Att}$ as

$$\mathbf{p}_{ho}^{Att} = \sigma(\mathbf{f}_{ho}^{Att}). \tag{10}$$

In stacked generalization, a meta-learner is able to learn classification boundaries based on confidence scores. However, the confidence scores produced by sub-models are typically not actual probabilities or are not well calibrated [25]. In this study, we use the feature vector $\mathbf{f}_{ho}^{Att}$ as an input for the meta-learner, as discussed in Section III-D, instead of directly using the confidence score vector $\mathbf{p}_{ho}^{Att}$.

As discussed in [29], $\mathbf{a}_{ho}$ is also used as an attention mechanism to refine visual features by multiplying two vectors.

$$\mathbf{a}_{ho}^{Ref} = \mathbf{a}_{ho} \otimes \mathbf{f}_{ho}^{Vis}, \tag{11}$$

$$\mathbf{f}_{ho}^{Ref} = \mathbf{W}^{Ref}(\mathbf{a}_{ho}^{Ref}), \tag{12}$$

where $\otimes$ denotes element-wise multiplication and $\mathbf{f}_{ho}^{Ref}$ is a spatially refined feature vector of size M.

The refined feature vector $\mathbf{f}_{ho}^{Ref}$ is utilized to predict the probabilities of action classes $\mathbf{p}_{ho}^{Ref}$ and the interaction proposal score i_{ho} for the human and object pair ho as follows:

$$\mathbf{p}_{ho}^{Ref} = \sigma(\mathbf{f}_{ho}^{Ref}), \tag{13}$$

$$f_{ho}^{IP} = \mathbf{W}^{IP}(\mathbf{a}_{ho}^{Ref}), \tag{14}$$

$$i_{ho} = \sigma(f_{ho}^{IP}), \tag{15}$$

where f_{ho}^{IP} is a scalar for the interaction proposal probability, which can be used as an input for the meta-learner.

3) GRAPH CONVOLUTIONAL INTERACTION BRANCH

This branch generates effective features using a graph convolutional network [14] in which humans and objects are represented as nodes and their relationships are represented as edges, as discussed in [29].

Given the visual features $\mathbf{f}_{ho}$ and $\mathbf{g}_o$ as nodes, the graph feature vectors $\mathbf{f}_{ho}{}'$ and $\mathbf{f}_{oh}{}'$ are defined as follows:

$$\mathbf{f}_{ho}{}' = \mathbf{f}_{ho} + \Sigma_{o=1}^{O} \alpha_{ho} \mathbf{W}_{oh}(\mathbf{g}_o), \tag{16}$$

$$\mathbf{f}_{oh}{}' = \mathbf{g}_o + \Sigma_{h=1}^{H} \alpha_{oh} \mathbf{W}_{ho}(\mathbf{f}_{ho}), \tag{17}$$

where α_{ho} represents the adjacency between h and o. $\mathbf{W}_{oh}$ and $\mathbf{W}_{ho}$ are mapping functions that project object features into the human feature space and vice versa, respectively. Adjacency values are defined by the interaction score between a human pose and object pair as

$$\alpha_{ho} = \alpha_{oh} = i_{ho}. \tag{18}$$

By pairing graph features and projecting them using a projection matrix $\mathbf{W}^{Graph}$, the graph feature vector $\mathbf{f}_{ho}^{Graph}$ for action prediction is calculated as

$$\mathbf{f}_{ho}^{Graph} = \mathbf{W}^{Graph}(\mathbf{a}_{ho} \otimes \mathbf{W}^{Concat}(\mathbf{f}_{ho}{}' \oplus \mathbf{f}_{oh}{}')), \tag{19}$$

where $\mathbf{W}^{Concat}$ a fully connected layer. $\mathbf{f}_{ho}^{Graph}$ can be used as an input for the meta-learner and the action class probability vector $\mathbf{p}_{ho}^{Graph}$ is defined as

$$\mathbf{p}_{ho}^{Graph} = \sigma(\mathbf{f}_{ho}^{Graph}), \tag{20}$$

where $\sigma(\cdot)$ is a sigmoid function.

All of the action predictions from the three branches and an interaction proposal score are combined by multiplying the probabilities as follows:

$$\mathbf{p}_{ho} = (\mathbf{p}_{ho}^{Att} \otimes \mathbf{p}_{ho}^{Ref} \otimes \mathbf{p}_{ho}^{Graph}) \times i_{ho}, \tag{21}$$

where $\mathbf{p}_{ho}$ is the final prediction vector for HOI detection.

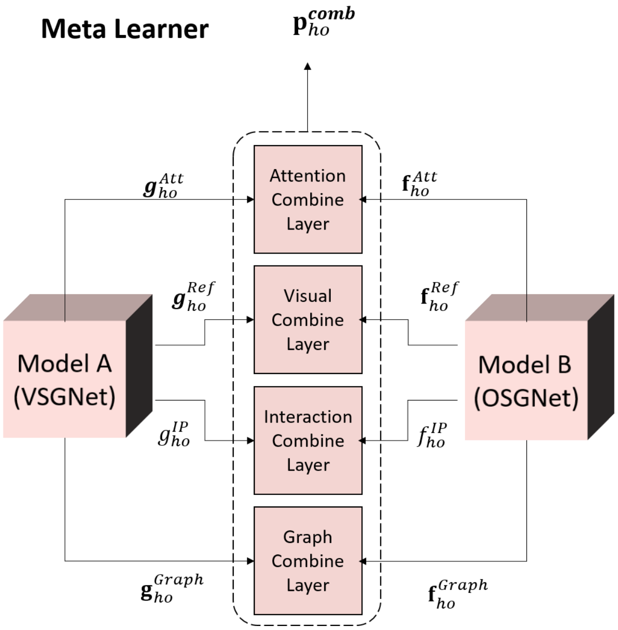


FIGURE 5. Meta-learner. The proposed scheme simultaneously trains the sub-models and meta-learner, which allows for on-the-fly stacked generalization.

D. LEVEL 1: ON-THE-FLY STACKED GENERALIZATION

1) STACKED GENERALIZATION

The meta-learner at level 1 learns how to combine the outputs of sub-models to maximize generalization accuracy. We designed the meta-learner with four layers of convolutional modules $\mathcal{C}^{\{\}}$ using the *ReLU* activation function, as shown in Figure 5 and Table 5. The inputs for the meta-learner are two sets of three action class features and an interaction feature generated by sub-models A and B.

$$\mathbf{f}_{ho}^{Att-comb} = \mathcal{C}^{Att}(\mathbf{g}_{ho}^{Att} \oplus \mathbf{f}_{ho}^{Att}), \tag{22}$$

$$\mathbf{f}_{ho}^{Ref-comb} = \mathcal{C}^{Ref}(\mathbf{g}_{ho}^{Ref} \oplus \mathbf{f}_{ho}^{Ref}), \tag{23}$$

$$\mathbf{f}_{ho}^{Graph-comb} = \mathcal{C}^{Graph}(\mathbf{g}_{ho}^{Graph} \oplus \mathbf{f}_{ho}^{Graph}), \tag{24}$$

$$f_{ho}^{IP-comb} = \mathcal{C}^{IP}(g_{ho}^{IP} \oplus f_{ho}^{IP}). \tag{25}$$

Finally, we obtain three combined action class probabilities and an interaction score by normalizing the four combined features to a range of zero to one using a sigmoid function $\sigma(\cdot)$ as follows:

$$\mathbf{p}_{ho}^{Att-comb} = \sigma(\mathbf{f}_{ho}^{Att-comb}), \tag{26}$$

$$\mathbf{p}_{ho}^{Ref-comb} = \sigma(\mathbf{f}_{ho}^{Ref-comb}), \tag{27}$$

$$\mathbf{p}_{ho}^{Graph-comb} = \sigma(\mathbf{f}_{ho}^{Graph-comb}), \tag{28}$$

$$i_{ho}^{comb} = \sigma(f_{ho}^{IP-comb}). \tag{29}$$

For the final action prediction vector $\mathbf{p}_{ho}^{comb}$ at level 1, all three action predictions and an interaction proposal score are combined by multiplying the probabilities as follows:

$$\mathbf{p}_{ho}^{comb} = (\mathbf{p}_{ho}^{Att-comb} \otimes \mathbf{p}_{ho}^{Ref-comb} \otimes \mathbf{p}_{ho}^{Graph-comb}) \times i_{ho}^{comb}. \tag{30}$$

2) ON-THE-FLY TRAINING LOSS

In the traditional stacked generalization framework, the sub-models at level 0 and meta-learner at level 1 are trained independently. In contrast, the proposed scheme simultaneously trains the sub-models and meta-learner, which allows for on-the-fly stacked generalization. We achieve this by combining the losses of each sub-model with that of the meta-learner.

$$\mathcal{L} = \lambda_1 \cdot \mathcal{L}_{comb} + \lambda_2 \cdot \mathcal{L}_A + \lambda_3 \cdot \mathcal{L}_B, \tag{31}$$

where $\mathcal{L}_{\{\}}$ represents binary cross entropy loss [8] for multi-label action classification and λ_i represents the weight factors for each term during on-the-fly training. The inputs for the three loss functions are $\mathbf{p}_{ho}^{comb}$, $\mathbf{p}_{ho}^{A}$, and $\mathbf{p}_{ho}^{B}$, respectively. The superscripts "*A*" and "*B*" denote the action prediction probabilities $\mathbf{p}_{ho}^{\{\}}$ obtained from sub-models *A* and *B*, respectively.

IV. EXPERIMENTS

A. DATASETS AND EVALUATION METRICS

For our experiments, we adopted the widely used V-COCO dataset [9] and HICO-DET dataset [4] as HOI benchmarks. The V-COCO dataset was generated by adding annotations for 29 actions (interactions) to 10,346 images containing 16,199 people based on the COCO dataset [19]. Among the 29 actions, four actions do not have pair relationships with objects and one action (point) has only 21 samples. Therefore, similar to previous HOI detection studies, for our experiment on the V-COCO dataset, we only report the performance of a total of 24 classes.

The HICO-DET dataset contains interactions between people and objects with box-level annotations. It contains a total of 47,776 images with 117 actions. The object categories correspond to 80 classes in the COCO dataset [19]. The total number of possible triplets (i.e., {human, object, action}) is 9,360 (=1×117×80), but this the dataset, only 600 categories are defined and have training data. According to the numbers of training samples in each category, 138 categories were defined as rare categories and the remaining 462 categories were defined as non-rare categories.

The evaluation metric for V-COCO is defined as the role average precision (AP) value according to the original paper on this data [9]. In the role AP calculation, two conditions must be satisfied: 1) the bounding boxes of the person and object must have an IoU greater than 0.5 with the ground-truth box and 2) the predicted interaction class label must be the same as the ground-truth interaction label. For the official metric [9], two scenarios are introduced for role AP evaluation. If an object does not exist (i.e., only humans are present), in scenario 1, a prediction is correct if the corresponding object box is empty. In scenario 2, a prediction is correct if the corresponding object box is ignored. This makes scenario 1 much more difficult than scenario 2.

The performance of HICO-DET was measured similarly to that of scenario 1 for V-COCO and is reported in terms of mean AP for all, rare, and non-rare categories.

B. IMPLEMENTATION DETAILS

Regarding our experimental implementation, if no additional explanation is given, we follow the process detailed in [29]. We extract two classification-based feature maps $\mathbf{G}^g$ and $\mathbf{G}^l$ from the two residual blocks prior to the final residual block in a Resnet-152 model [11] that was pre-trained on the ImageNet dataset [36] for a classification task, as shown in Figure 2a. For the human-pose-estimation-based feature map $\mathbf{F}_h$, we use the feature extractor for HRNet [28], which was pre-trained on the COCO dataset [19] for a pose estimation task, as shown in Figure 2b. The feature extractors (*i.e.*, ResNet-152 and HRNet) were not fine-tuned during training, and the feature maps $\mathbf{G}^g$, $\mathbf{G}^l$, and $\mathbf{F}_h$ serve as inputs for the rest of the HOI relationship networks in the two sub-models.

By using ROI pooling, 10×10 feature maps were extracted for all humans, objects, human poses, and five different body parts in the input feature maps. This operation is followed by a residual block and GAP, and three feature vectors of size $R = 1024$ for humans, objects, and contexts are obtained. We also obtain three feature vectors of size $P = 32$ for a human pose and size $Q = (32 + 512) \times 5$ for five human body parts, as well as a coordinate encoding feature vector of size $S = 20$. These features are fed into the remainder of the network. We use $64 \times 64 \times 2$ and $64 \times 64 \times 6$ binary inputs for the spatial attention branches in sub-models A and B, respectively. As discussed in [29], all input feature vectors are projected into a $D = 512$-dimensional space. For final classification, one linear layer is applied to all branches. The detailed configurations of the residual blocks $Res^{\{\}}_{\{\}}$, convolution block in the spatial branch $Conv^{Spat}$, projection layer $\mathcal{W}^{Vis}$ in Equation 6, and combination layers $\mathcal{C}^{\{\}}$ in the meta-learner are provided in Tables 2, 3, 4, and 5, respectively.

TABLE 2. Configuration of the residual blocks discussed in Section III. [·] denotes the kernel size and number of input and output channels. Batch normalization [12] follows each *Conv* layer in $Res^{\{\}}_{\{\}}$. See [11] for additional details.

Layer Name	$Res^{\{\}}_{\{\}}$
Conv	$[1 \times 1, 1024, 512]$ stride=1
Conv	$[1 \times 1, 512, 512]$ stride=1
Conv	$[1 \times 1, 512, 1024]$ stride=1
ReLU	-

To generate bounding boxes $\mathbf{x}_h^k$ based on body keypoint locations, we followed the method discussed in [2]. This approach simply sets the width and height of a box by adding a margin derived from the maximum and minimum values of the spatial distribution of body keypoint locations.

TABLE 3. Configuration of $Conv^{Spat}$ discussed in Section III-C. [·] denotes the kernel size and numbers of input and output channels.

Layer Name	$Conv^{Spat}$
Conv	$[5 \times 5, 6, 64]$ stride=1
Max pooling	$[2 \times 2]$ stride=1
Conv	$[5 \times 5, 63, 32]$ stride=1
Max pooling	$[2 \times 2]$ stride=1
GAP	-

TABLE 4. Configuration of a sequence of two fully connected layers with a *ReLU* function denoted as $\mathcal{W}^{\{\}}$ in Equation 6. *D*, *P*, *Q*, and *S* are the feature dimensions for concatenation, as discussed in Section IV-B.

Name	#Input	#Output
Linear	$2 \times R + P + Q + S$	$3 \times R + S$
ReLU	-	-
Linear	$3 \times R + S$	R
Linear	R	D
ReLU	-	-

TABLE 5. Configuration of the combination layers $\mathcal{C}^{\{\}}$ discussed in Section III-D. #Input and #Output denote the number of input and output channels, respectively. *M* is the number of HOI classes. For the interaction proposal probability, *M* is set to one.

Name	#Input	#Output
Linear	$2 \times M$	$4 \times M$
ReLU	-	-
Linear	$4 \times M$	M

For the V-COCO dataset, the batch size and initial learning rate were set to 8 and 0.01, respectively. We set the weight decay to 0.0001 and used a momentum of 0.9 for stochastic gradient descent optimization. For more efficient training, the learning rate was increased to 0.01 for all layers except for the spatial attention branch between epochs 9 and 21 among a total of 50 epochs. For the HICO-DET dataset, we used the same hyperparameters as those used for V-COCO. We trained the networks for 80 epochs on the HICO-DET training set. The rest of the details are the same as those for the VSGNet (see [29] for additional details).

We measured training and inference times on a single NVIDIA Titan RTX GPU after fixing the numbers of detected persons and objects to one because the training time and inference time are determined by the numbers of detected persons and objects. The training and inference times in one epoch for the V-COCO dataset were about 4.5 minutes and 3.8 minutes, respectively. For the HICO-DET dataset, the training and inference times in one epoch were about 33.7 minutes and 7.6 minutes, respectively. The number of parameters is approximately 135M and the number of GFLOPs is 40.6 for OSGNet.

C. COMPARISONS TO STATE-OF-THE-ART METHODS

We compared the performance of our method to those of state-of-the-art HOI detection methods using the V-COCO and HICO-DET benchmark datasets. Table 6 presents performance comparisons on the V-COCO dataset. Our method outperforms the state-of-the-art methods (InteractNet [7], Kolesnikov *et al.* [15], GPNN [24], iCAN [5], Interactiveness [17], and ACP [13]) by a mean average precision (mAP) of 53.43. Additionally, our model outperforms the baseline (VSGNet) by approximately 2% when using the same bounding boxes.

TABLE 6. Comparison results for the V-COCO [9] test set for scenarios 1 and 2.

V-COCO	mAP (Sc1)	mAP (Sc2)
InteractNet [7]	40.0	47.98
Kolesnikov *et al.* [15]	41.0	-
GPNN [24]	44.0	-
iCAN [5]	45.3	52.4
Interactiveness [17]	47.8	-
RPNN [35]	47.53	-
PMFNet [30]	52.0	-
ACP [13]	52.98	-
Baseline-VSGNet [29]	51.76	57.03
Proposed (OSGNet)	**53.43**	**58.68**

Table 7 presents performance comparisons on the HICO-DET dataset. Our method outperforms all of the state-of-the-art methods, except for PPDM [18]. In PPDM, the object detection network was also trained to be suitable for HOI problems and exhibits high performance for non-rare classes. However, PPDM exhibits low performance for rare classes, which is an important problem in HOI detection based on common data imbalances. Our method exhibits similar performance compared to PPDM overall. However, for rare classes, the proposed method exhibits higher performance than PPDM.

TABLE 7. Comparison of results for the HICO-DET [4] test set. The proposed method outperforms the baseline. Notably, DJ-RN uses additional 3D human pose features. Excluding DJ-RN, the proposed OSGNet exhibits the best performance for rare classes.

HICO-DET	Full	Rare	Non-Rare
Shen *et al.* [27]	6.46	4.24	7.12
HO-RCNN [4]	7.81	5.37	8.54
InteractNet [7]	9.94	7.16	10.77
GPNN [24]	13.11	9.34	9.34
iCAN [5]	14.84	10.45	16.15
Interactiveness [17]	17.03	13.42	18.11
No-Frills [10]	17.18	12.17	18.68
RPNN [35]	17.35	12.78	18.71
PMFNet [30]	17.46	15.65	18.00
Peyre *et al.* [23]	19.40	14.60	20.90
ACP [13]	20.59	15.92	21.98
DJ-RN [16]	21.34	**18.53**	22.18
S2D [16]	19.98	16.97	20.88
S3D [16]	12.41	13.08	12.21
SJoint [16]	20.61	17.01	21.69
PPDM [18]	**21.73**	13.78	**24.10**
Baseline-VSGNet [29]	19.80	16.05	20.91
Proposed (OSGNet)	***21.40***	***18.12***	***22.38***

Among the rare class performance results for the HICO-DET dataset, DJ-RN achieves the best performance. However, DJ-RN uses detailed 3D body representations with spatial volumes, which significantly increases model complexity. While the proposed method operates in an end-to-end manner using only 2D image inputs, DJ-RN first extracts the 2D poses of the body, face, and hands, after which SMPLify-X [22] is applied to obtain 3D body information for inputs. Overall, the performance of the proposed method using an on-the-fly stacking structure is comparable to that of DJ-RN, which improves generalization using 3D information.

TABLE 8. Ablation studies for two branches of sub-Model *A* and sub-Model *B*, as well as the complete OSGNet.

HOI Class	Sub-Model A	Sub-Model B	Comb.
hold-obj	48.20	49.50	**51.29**
sit-instr	28.48	28.00	**30.13**
ride-instr	65.69	70.05	**71.83**
look-obj	43.43	45.28	**45.71**
hit-instr	76.81	76.15	**78.65**
hit-obj	48.40	47.77	**49.45**
eat-obj	36.88	38.56	**38.82**
eat-instr	**6.26**	5.11	6.14
jump-instr	51.27	52.03	**52.94**
lay-instr	24.18	21.19	**24.33**
talk_on_phone	61.06	61.63	**62.54**
carry-obj	37.48	40.48	**41.61**
throw-obj	43.29	45.79	**48.61**
catch-obj	44.10	45.02	**48.77**
cut-instr	**48.96**	47.72	48.93
cut-obj	39.08	40.73	**41.56**
work_on_comp	65.07	66.42	**67.16**
ski-instr	49.27	51.90	**54.20**
surf-instr	81.21	82.03	**82.41**
skateboard-instr	87.97	88.14	**89.09**
drink-instr	46.65	49.86	**50.40**
kick-obj	73.67	76.09	**76.55**
read-obj	35.53	40.74	**41.30**
snowboard-instr	78.72	79.52	**79.86**
Average	50.90	52.08	**53.43**

D. ANALYSIS OF BRANCH PERFORMANCE

For our ablation study, we evaluated the performance of each branch of OSGNet (sub-model A and sub-model B). As reported in Table 8, the performance of sub-model B is higher than that of sub-model A for most classes in the V-COCO dataset. Therefore, for the HOI detection problem, we hypothesize that human poses and part features can provide a better understanding of detailed interactions. However, for the classes "eat-instr" and "cut-instr," sub-model A exhibits slightly better performance. The reason for this is that the "eat-obj" class represents an interaction in which a person eats an object, whereas the "eat-instr" class judges whether a person uses an instrument to eat. Therefore, the object class itself is a more important cue compared to detailed relationships between humans and objects. Additionally, because OSGNet enables the combination of global representations (sub-model A) and local representations (sub-model B) for one problem, it achieves a performance improvement of nearly 3% compared to sub-model A alone.

E. ON-THE-FLY TRAINING LOSS

In this section, we analyze the effects of the on-the-fly training loss in Equation (31). We can adjust the sub-model and meta-learner losses by using the parameter λ_i to balance stacked generalization. As shown in Table 9, the mAP values

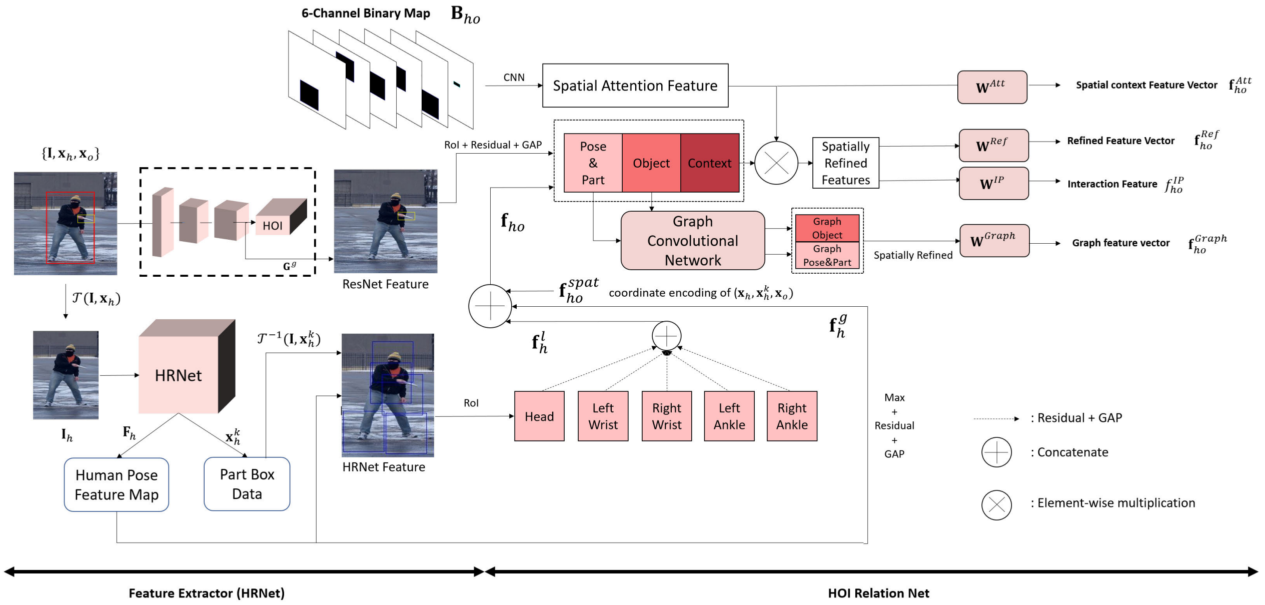

6-Channel Binary Map $\mathbf{B}_{ho}$
CNN
Spatial Attention Feature
$\mathbf{W}^{Att}$
Spatial context Feature Vector $\mathbf{f}_{ho}^{Att}$
$\{\mathbf{I}, \mathbf{x}_h, \mathbf{x}_o\}$
RoI + Residual + GAP
Pose & Part
Object
Context
Spatially Refined Features
$\mathbf{W}^{Ref}$
Refined Feature Vector $\mathbf{f}_{ho}^{Ref}$
$\mathbf{W}^{IP}$
Interaction Feature f_{ho}^{IP}
HOI
$\mathbf{G}^g$
ResNet Feature
$\mathbf{f}_{ho}$
Graph Convolutional Network
Graph Object
Graph Pose&Part
Spatially Refined
$\mathbf{W}^{Graph}$
Graph feature vector $\mathbf{f}_{ho}^{Graph}$
$\mathcal{T}(\mathbf{I}, \mathbf{x}_h)$
$\mathbf{f}_{ho}^{spat}$
coordinate encoding of $(\mathbf{x}_h, \mathbf{x}_h^k, \mathbf{x}_o)$
$\mathbf{f}_h^g$
$\mathbf{f}_h^l$
HRNet
$\mathcal{T}^{-1}(\mathbf{I}, \mathbf{x}_h^k)$
$\mathbf{I}_h$
$\mathbf{F}_h$
$\mathbf{x}_h^k$
Human Pose Feature Map
Part Box Data
HRNet Feature
RoI
Head
Left Wrist
Right Wrist
Left Ankle
Right Ankle
Max + Residual + GAP
: Residual + GAP
: Concatenate
: Element-wise multiplication
Feature Extractor (HRNet)
HOI Relation Net

FIGURE 6. Framework of OSGNet-p.

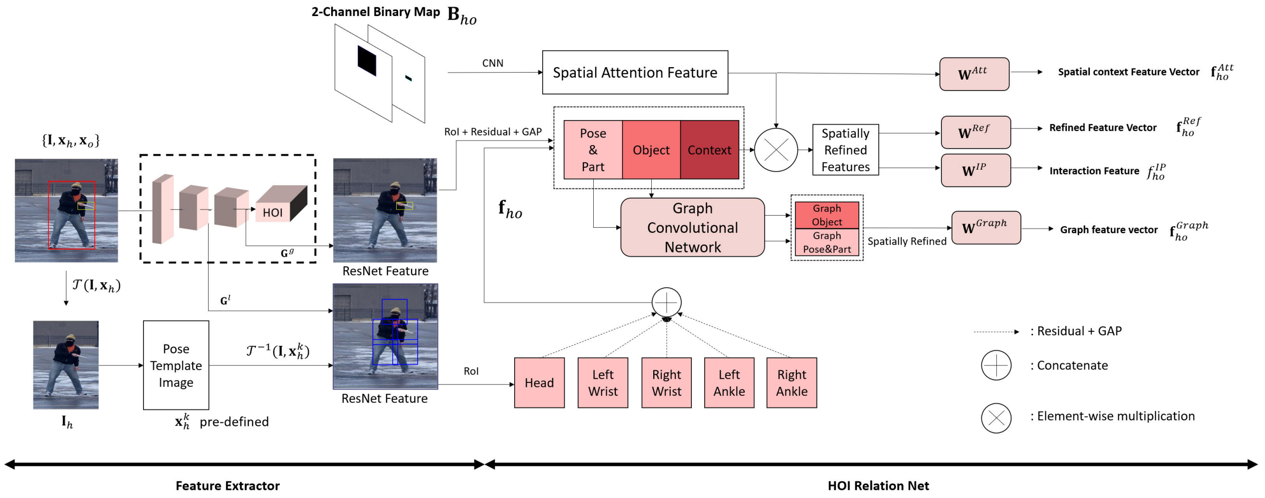

2-Channel Binary Map $\mathbf{B}_{ho}$
CNN
Spatial Attention Feature
$\mathbf{W}^{Att}$
Spatial context Feature Vector $\mathbf{f}_{ho}^{Att}$
$\{\mathbf{I}, \mathbf{x}_h, \mathbf{x}_o\}$
RoI + Residual + GAP
Pose & Part
Object
Context
Spatially Refined Features
$\mathbf{W}^{Ref}$
Refined Feature Vector $\mathbf{f}_{ho}^{Ref}$
$\mathbf{W}^{IP}$
Interaction Feature f_{ho}^{IP}
HOI
$\mathbf{G}^g$
ResNet Feature
$\mathbf{f}_{ho}$
Graph Convolutional Network
Graph Object
Graph Pose&Part
Spatially Refined
$\mathbf{W}^{Graph}$
Graph feature vector $\mathbf{f}_{ho}^{Graph}$
$\mathcal{T}(\mathbf{I}, \mathbf{x}_h)$
$\mathbf{G}^l$
Pose Template Image
$\mathcal{T}^{-1}(\mathbf{I}, \mathbf{x}_h^k)$
$\mathbf{I}_h$
$\mathbf{x}_h^k$ pre-defined
ResNet Feature
RoI
Head
Left Wrist
Right Wrist
Left Ankle
Right Ankle
: Residual + GAP
: Concatenate
: Element-wise multiplication
Feature Extractor
HOI Relation Net

FIGURE 7. Framework of OSGNet-e.

of the trained models using only $\mathcal{L}_A$ and $\mathcal{L}_B$ are recorded as 52.35 and 52.50 for scenario 1, respectively. In the case of scenario 2, the trained models using only $\mathcal{L}_A$ and $\mathcal{L}_B$ yield mAP values of 57.58 and 57.70, respectively. When the combined loss for on-the-fly training is applied, performance is improved compared using the loss of either sub-model alone in all cases. When the weight of the combined loss is equal to two, the highest performance is achieved.

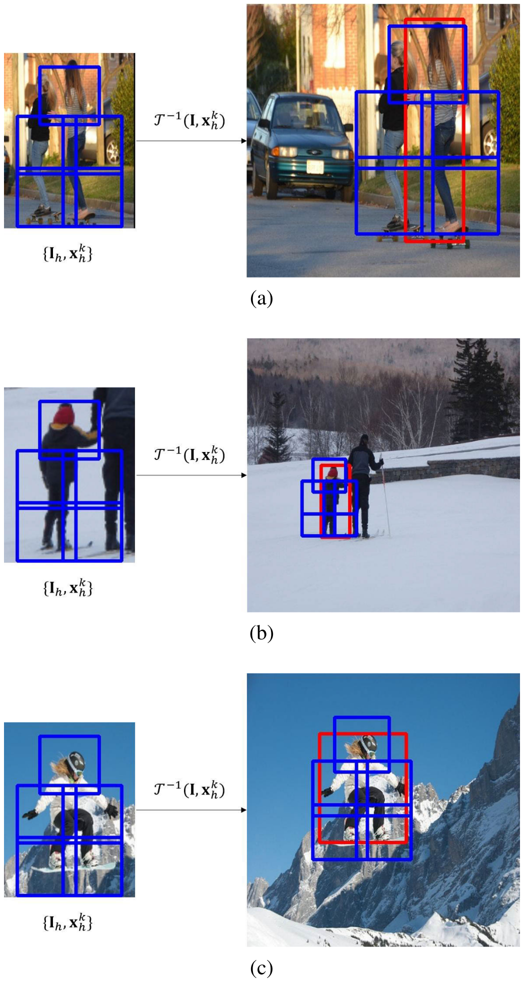
TABLE 9. Analysis of stacked generalization.

Training Loss	mAP (Sc1)	mAP (SC2)
$\mathcal{L}_A$	52.35	57.58
$\mathcal{L}_B$	52.50	57.70
$1 \times \mathcal{L}_{comb} + \mathcal{L}_A + \mathcal{L}_B$	52.83	57.97
$2 \times \mathcal{L}_{comb} + \mathcal{L}_A + \mathcal{L}_B$	**53.43**	**58.68**
$3 \times \mathcal{L}_{comb} + \mathcal{L}_A + \mathcal{L}_B$	53.08	58.26
$4 \times \mathcal{L}_{comb} + \mathcal{L}_A + \mathcal{L}_B$	52.96	58.16
$5 \times \mathcal{L}_{comb} + \mathcal{L}_A + \mathcal{L}_B$	52.89	58.13

F. TWO VARIANT STRUCTURES OF OSGNet

We designed two variant architectures called OSGNet-p (Figure 6) and OSGNet-e (Figure 7) to analyze the effects of human poses and low-level features on the proposed OSGNet.

For OSGNet-p, we removed the usage of $\mathbf{G}^l$ for the part features extracted from ResNet by comparing Figures 4 and 6. In other words, in the original OSGNet, low-level


$\mathcal{T}^{-1}(\mathbf{I},\mathbf{x}_h^k)$
$\{\mathbf{I}_h,\mathbf{x}_h^k\}$
$\mathcal{T}^{-1}(\mathbf{I},\mathbf{x}_h^k)$
$\{\mathbf{I}_h,\mathbf{x}_h^k\}$
$\mathcal{T}^{-1}(\mathbf{I},\mathbf{x}_h^k)$
$\{\mathbf{I}_h,\mathbf{x}_h^k\}$

(a)

(b)

(c)

FIGURE 8. Examples of emulated bounding boxes for OSGNet-e, including human keypoint bounding boxes for human-centric images and inverse affine transformation results for input images.

TABLE 10. Comparison of class-wise average precision values to those of the state-of-the-art methods for V-COCO scenario 1. The "obj" and "instr" tags refer to objects and instruments [9], respectively. OSGNet is written as OSG to reduce the size of the table.

HOI Class	VSGNet	OSG	OSG-p	OSG-e
hold-obj	48.27	51.29	**52.73**	49.92
sit-instr	29.9	30.13	**30.90**	29.83
ride-instr	70.84	71.83	**73.02**	71.03
look-obj	42.78	45.71	**46.48**	43.69
hit-instr	76.08	**78.65**	78.35	76.79
hit-obj	48.60	**49.45**	49.17	49.25
eat-obj	38.30	38.82	40.97	**41.10**
eat-instr	6.30	6.14	6.14	**6.52**
jump-instr	52.66	52.94	**53.18**	52.37
lay-instr	21.66	24.33	24.57	**24.68**
talk_on_phone	62.23	62.54	62.58	**62.72**
carry-obj	39.09	41.61	**43.07**	39.08
throw-obj	45.12	48.61	47.40	**48.75**
catch-obj	44.84	48.77	**49.01**	46.64
cut-instr	46.78	48.93	**49.59**	47.95
cut-obj	36.58	**41.56**	38.36	40.25
work_on_comp	64.60	67.16	**68.39**	65.49
ski-instr	50.59	**54.20**	52.16	51.37
surf-instr	82.22	**82.41**	82.11	**82.41**
skateboard-instr	87.80	89.09	**89.87**	89.52
drink-instr	**54.41**	50.40	54.18	51.34
kick-obj	69.85	**76.55**	75.85	68.39
read-obj	**42.83**	41.30	41.73	39.20
snowboard-instr	79.90	79.86	**80.50**	79.46
Average	51.76	53.43	**53.76**	52.40

TABLE 11. Comparison of results for the HICO-DET [4] test set.

HICO-DET	Full	Rare	Non-Rare
Baseline-VSGNet [29]	19.80	16.05	20.91
OSGNet	**21.40**	**18.12**	**22.38**
OSGNet-p	20.93	17.35	22.00
OSGNet-e	20.59	17.19	21.61

part features are extracted from both ResNet and HRNet, but OSGNet-p only extracts part features from HRNet. In the case of OSGNet-e, we emulate predefined keypoints[1] in the affine-transformed image (we call this a pose template), resulting in keypoint locations that are the same across all transformed images $\mathbf{I}_h$. We also use a two-channel binary map B_{ho}, as shown in Figure 7. However, the sizes of bounding boxes based on the emulated keypoints can be adjusted to fit the size of a person in an input image $\mathbf{I}$ by inversely transforming them to match the input image, as shown in Figure 8. In this manner, we can extract low-level features around a human for the proposed OSGNet, even though the keypoint locations are inaccurate in input images.

Tables 10 and 11 present performance comparisons between VSGNet, OSGNet, OSGNet-p, and OSGNet-e on the V-COCO and HICO-DET datasets, respectively. In the case of the V-COCO dataset, OSGNet-p exhibits slightly better performance than OSGNet. For the HICO-DET dataset, OSGNet exhibits better performance than OSGNet-p. We believe that in the V-COCO dataset, human images tend to include most of the body and overlap with objects is not severe. In the HICO-DET dataset, there are many images in which only a part of a person appears and much of the person is obscured by an object. Therefore, using not only the features from HRNet for humans, but also the features from ResNet for objects, is helpful for improving performance. Surprisingly, OSGNet-e outperforms VSGNet on both benchmark datasets. This means that the proposed framework itself, even without human pose features, can help improve the generalization accuracy.

G. QUALITATIVE RESULTS

Figures 9 and 10 present the qualitative results of HOI detection. The top-three class labels from sub-model A (VSGNet) are presented above the images and those from OSGNet are presented below the images. The baseline structure utilizes the global relationship between a person and object for HOI detection, but its results are not satisfactory when part or local information is important. However, because the proposed OSGNet utilizes both global and local information

[1] Based on an image with dimensions of 192×256 pixels ($width \times height$), five predefined keypoints (x, y) are defined at (95, 62), (129, 134), (62, 134), (129, 210), and (62, 210).

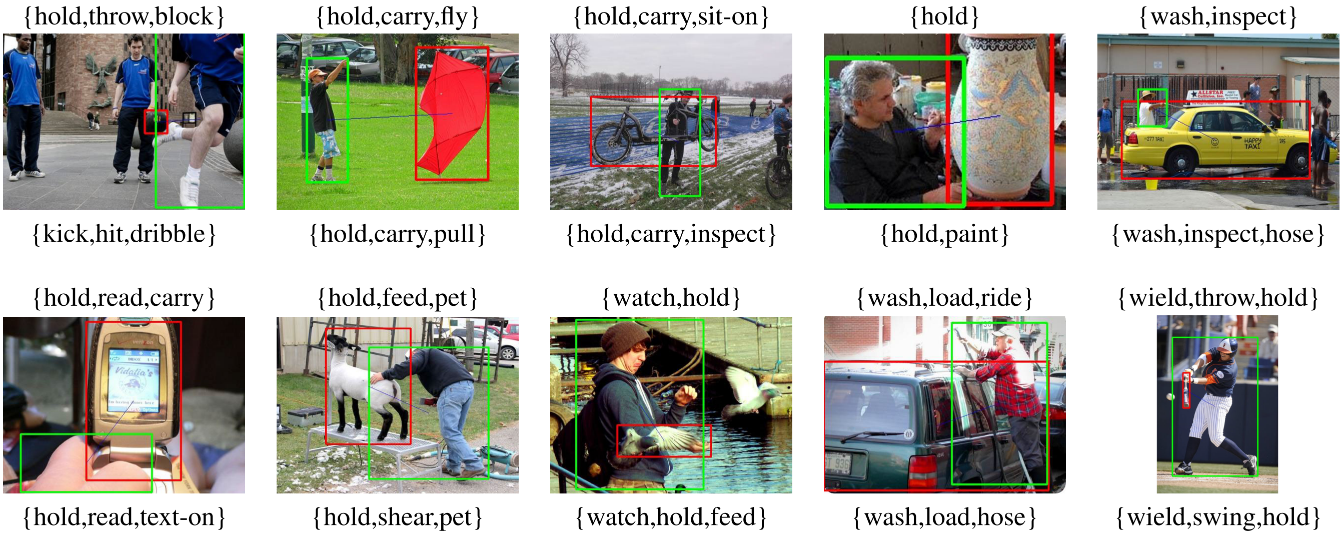


FIGURE 9. **Qualitative results for the HICO-DET dataset. The top-three classes are extracted from sub-model *A* (above the image) and OSGNet (below the image). Through the stacked generalization of OSGNet, the estimated classes are modified by focusing on both global and local features. If the probability is too low or the class is no-interaction, we omit it in {}.**

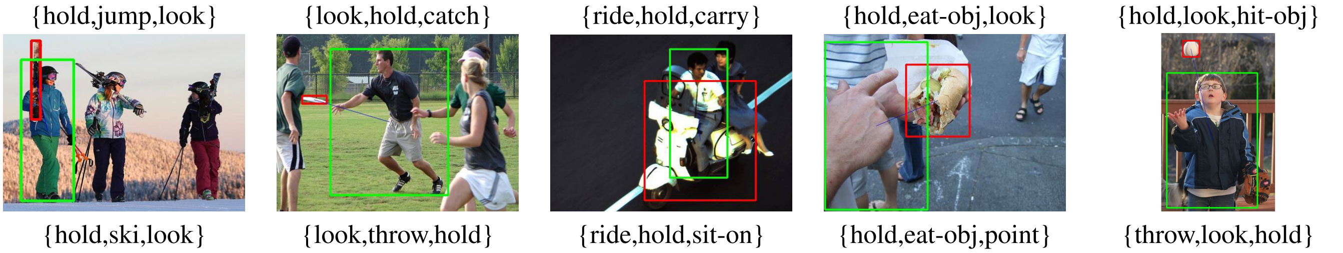


FIGURE 10. **Qualitative results for the V-COCO dataset. The top-three classes are extracted from sub-model *A* (above the image) and OSGNet (below the image). Through the stacked generalization of OSGNet, the estimated classes are modified by focusing on both global and local features.**

through on-the-fly stacked generalization, it corrects estimated classes and improves HOI performance, as shown in Figures 9 and 10.

V. CONCLUSION AND DISCUSSIONS

Although there is substantial evidence that human pose information is helpful for inferring relationships between people and objects, most HOI detection methods combine image and human pose information through a single underlying model and do not leverage the advantages of using multiple diverse models. In this article, we proposed a novel framework for HOI detection called OSGNet. The proposed method uses different feature extraction models trained for different tasks, namely classification and human pose estimation. Simultaneously training the sub-models and meta-learner allows the sub-models to learn complementary information for each other. Additionally, the proposed OSGNet even exhibits improved performance in cases without human pose features. We believe that the proposed method can also be utilized for other strategies to enhance generalization accuracy by combining two different task-based sub-models. As future work, we plan to study more robust and efficient algorithms that can be utilized in real-world environments. To this end, we could consider reducing the dependence of the object detector and extending the proposed method to use consecutive frames of a video.

REFERENCES

[1] K. Bae, K. Yun, H. Kim, Y. Lee, and J. Park, "Anti-litter surveillance based on person understanding via multi-task learning," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2020, pp. 1–13.

[2] B. Bhandari, G. Lee, and J. Cho, "Body-part-aware and multitask-aware single-image-based action recognition," *Appl. Sci.*, vol. 10, no. 4, p. 1531, Feb. 2020.

[3] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and A. M. J. Black, "Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 561–578.

[4] Y.-W. Chao, Y. Liu, X. Liu, H. Zeng, and J. Deng, "Learning to detect human-object interactions," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 381–389.

[5] C. Gao, Y. Zou, and J. B. Huang, "ICAN: Instance-centric attention network for human-object interaction detection," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2018, pp. 1–14.

[6] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1440–1448.

[7] G. Gkioxari, R. Girshick, P. Dollar, and K. He, "Detecting and recognizing human-object interactions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8359–8367.

[8] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[9] S. Gupta and J. Malik, "Visual semantic role labeling," 2015, *arXiv:1505.04474*. [Online]. Available: http://arxiv.org/abs/1505.04474

[10] T. Gupta, A. Schwing, and D. Hoiem, "No-frills human-object interaction detection: Factorization, layout encodings, and training techniques," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9677–9685.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[12] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Mach. Learn. Res.*, 2015, pp. 448–456.

[13] D.-J. Kim, X. Sun, J. Choi, S. Lin, and I.-S. Kweon, "Detecting human-object interactions with action co-occurrence priors," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 718–736.

[14] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent.*, Sep. 2017, pp. 1–13.

[15] A. Kolesnikov, A. Kuznetsova, C. Lampert, and V. Ferrari, "Detecting visual relationships using box attention," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 1749–1753.

[16] Y.-L. Li, X. Liu, H. Lu, S. Wang, J. Liu, J. Li, and C. Lu, "Detailed 2D-3D joint representation for human-object interaction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10166–10175.

[17] Y.-L. Li, S. Zhou, X. Huang, L. Xu, Z. Ma, H.-S. Fang, Y. Wang, and C. Lu, "Transferable interactiveness knowledge for human-object interaction detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3585–3594.

[18] Y. Liao, S. Liu, F. Wang, Y. Chen, C. Qian, and J. Feng, "PPDM: Parallel point detection and matching for real-time human-object interaction detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 482–490.

[19] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C . L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 740–755.

[20] Z. Ma, P. Wang, Z. Gao, R. Wang, and K. Khalighi, "Ensemble of machine learning algorithms using the stacked generalization approach to estimate the warfarin dose," *PLoS ONE*, vol. 13, no. 10, Oct. 2018, Art. no. e0205872.

[21] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 1–8.

[22] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3D hands, face, and body from a single image," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10975–10985.

[23] J. Peyre, J. Sivic, I. Laptev, and C. Schmid, "Detecting unseen visual relations using analogies," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1981–1990.

[24] S. Qi, W. Wang, B. Jia, J. Shen, and S.-C. Zhu, "Learning human-object interactions by graph parsing neural networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 407–423.

[25] N. F. Rajani and R. Mooney, "Stacking with auxiliary features for visual question answering," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1, 2018, pp. 2217–2226.

[26] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[27] L. Shen, S. Yeung, J. Hoffman, G. Mori, and L. Fei-Fei, "Scaling human-object interaction recognition through zero-shot learning," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1568–1576.

[28] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5693–5703.

[29] O. Ulutan, A. S. M. Iftekhar, and B. S. Manjunath, "VSGNet: Spatial attention network for detecting human object interactions using graph convolutions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13617–13626.

[30] B. Wan, D. Zhou, Y. Liu, R. Li, and X. He, "Pose-aware multi-level feature network for human object interaction detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9469–9478.

[31] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, Jan. 1992.

[32] S. Xia, L. Gao, Y.-K. Lai, M.-Z. Yuan, and J. Chai, "A survey on human performance capture and animation," *J. Comput. Sci. Technol.*, vol. 32, no. 3, pp. 536–554, May 2017.

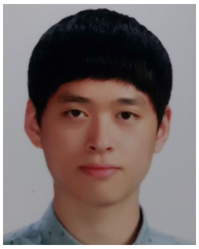
[33] K. Yun, Y. Kwon, S. Oh, J. Moon, and J. Park, "Vision-based garbage dumping action detection for real-world surveillance platform," *ETRI J.*, vol. 39, pp. 12–21, Jul. 2019.

[34] K. Yun, J. Park, and J. Cho, "Robust human pose estimation for rotation via self-supervised learning," *IEEE Access*, vol. 8, pp. 32502–32517, 2020.

[35] P. Zhou and M. Chi, "Relation parsing neural network for human-object interaction detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 843–851.

[36] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[37] Y.-L. Li, X. Liu, X. Wu, Y. Li, and C. Lu, "HOI analysis: Integrating and decomposing human-object interaction," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 1–12.



GEONU LEE is currently pursuing the bachelor's degree with the Department of Computer Engineering, Gachon University, Seongnam, South Korea. His research interests include deep learning, computer vision, and human action recognition.

KIMIN YUN received the B.S. and Ph.D. degrees from the Department of Electrical and Computer Engineering, Seoul National University, Seoul, South Korea, in 2010 and 2017, respectively. Since 2017, he has been working with the Artificial Intelligence Research Laboratory, Visual Intelligence Research Section, ETRI, Daejeon, South Korea. His current research interests include machine learning, computer vision, visual event analysis, moving object detection, and video analysis.

JUNGCHAN CHO (Member, IEEE) received the B.S. degree from the School of Electrical and Electronics Engineering, Chung-Ang University, Seoul, South Korea, in 2010, and the Ph.D. degree from the Department of Electrical and Computer Engineering, Seoul National University, Seoul, in 2016. From 2016 to 2019, he was a Senior Software Engineer with Samsung Electronics. He is currently an Assistant Professor with the Department of Software, Gachon University, Seongnam, South Korea. His research interests include deep learning, computer vision, and machine learning.