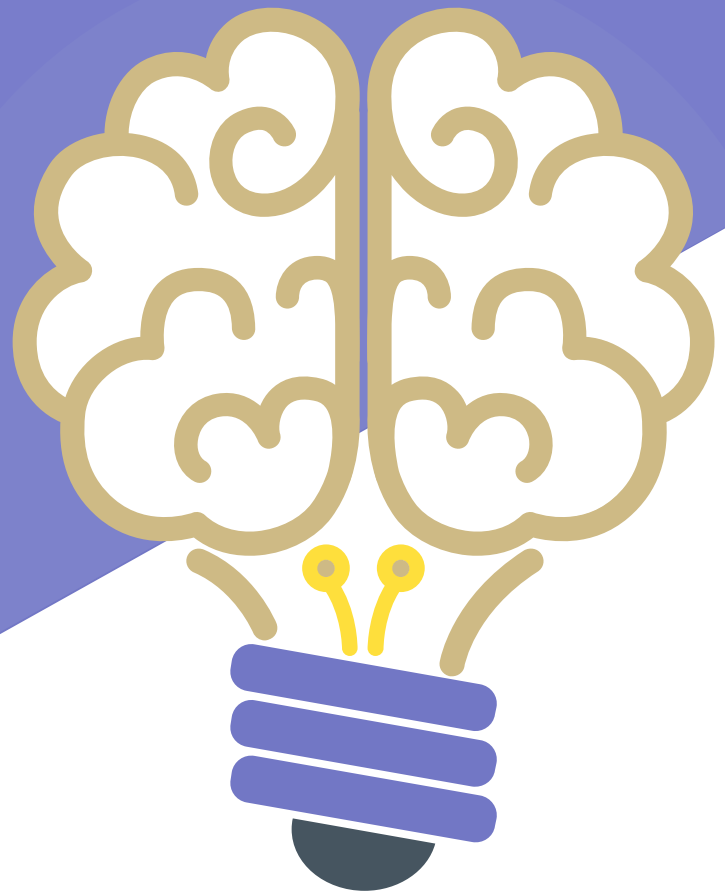
Pairwise Body-Part Attention for Recognizing Human-Object Interactions (ECCV 2018) PaStaNet: Toward Human Activity Knowledge Engine (CVPR2020) Masked Autoencoders Are Scalable Vision Learners（CVPR2022）



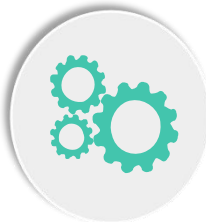
苏展（Zhan Su） 2022-1-14

Pairwise Body-Part Attention for Recognizing Human-Object Interactions

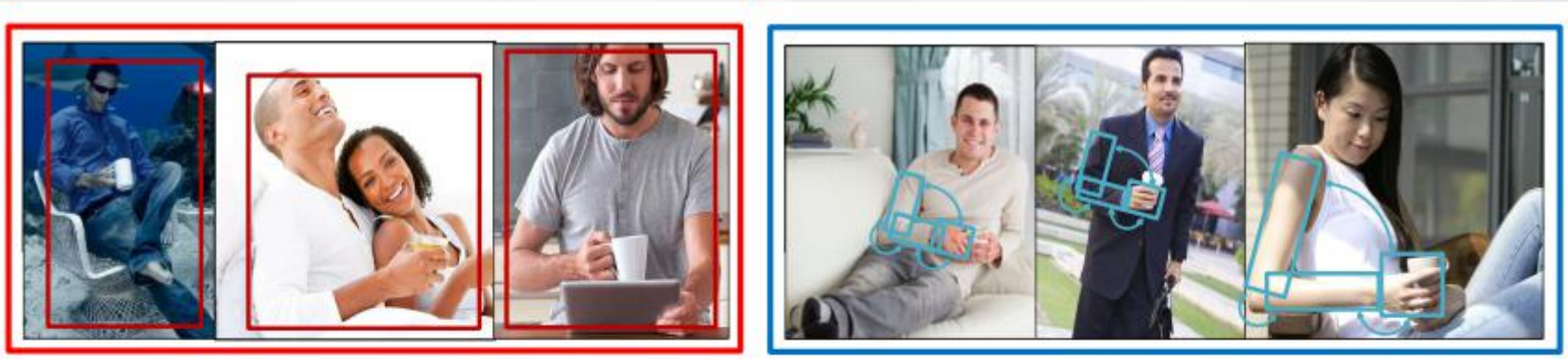
Hao-Shu Fang[1], Jinkun Cao[1], Yu-Wing Tai[2], and Cewu Lu[1]*

[1] Shanghai Jiao Tong University, China
fhaoshu@gmail.com, {caojinkun,lucewu}@sjtu.edu.cn
[2] Tencent YouTu Lab, China
yuwingtai@tencent.com

The Pairwise Body-Part Attention (PBPA) model solves the HOI recognition task. The PBPA model can focus the network on the important body parts of the human body in HOI and extract the relationship features between pairwise parts. Then fusion of these fine-grained features with the overall features can improve the recognition accuracy.



Pairwise Body-Part Attention for Recognizing Human-Object Interactions



(a) Conventional HOI recognition model

(b) Our model

Fig. 1. Given an image, a person holding a mug in his/her hand, conventional model (a) infers the HOI from the whole body feature. In contrast, our model (b) explicitly focuses on discriminative body parts and the correlations between objects and different body parts. In this example, the upper and lower arms which hold a mug form an acute angle across all of the above images.

The contribution of this paper mainly focuses on two points: 1. This paper considers the more fine-grained human body part features and the relationship features between pairwise parts, including appearance and spatial configuration. 2. They propose an attention model to select the most important features from many pairwise parts.

Pairwise Body-Part Attention for Recognizing Human-Object Interactions

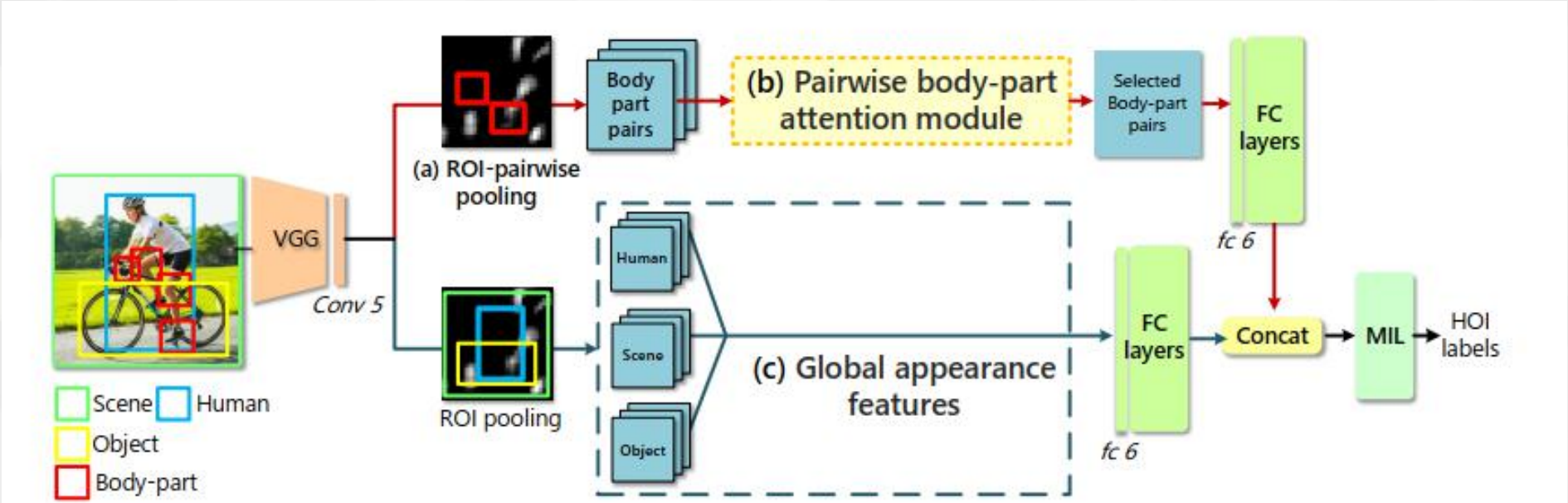
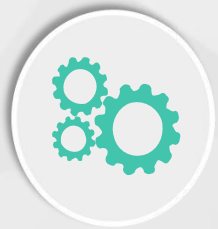
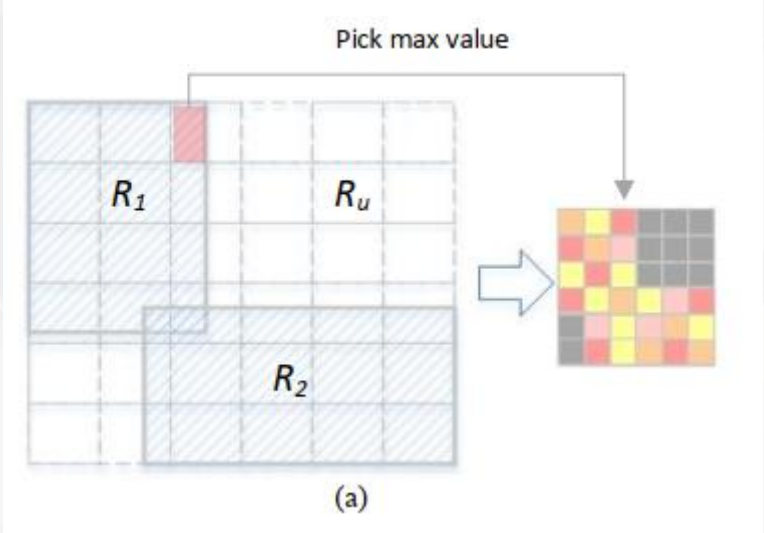


Fig. 2. Overview of our framework. The model first extracts visual features of human,object and scene from a set of proposals. We encode the features of different body parts and their pairwise correlations using ROI-pairwise pooling (a). Then our pairwise body-part attention module (b) will select the feature maps of those discriminative body-part pairs. The global appearance features (c) from the human, object and scene will also contribute to the final predictions. Following [29], we adopt MIL to address the problem of multi-person co-occurrence in an image. See text for more details.



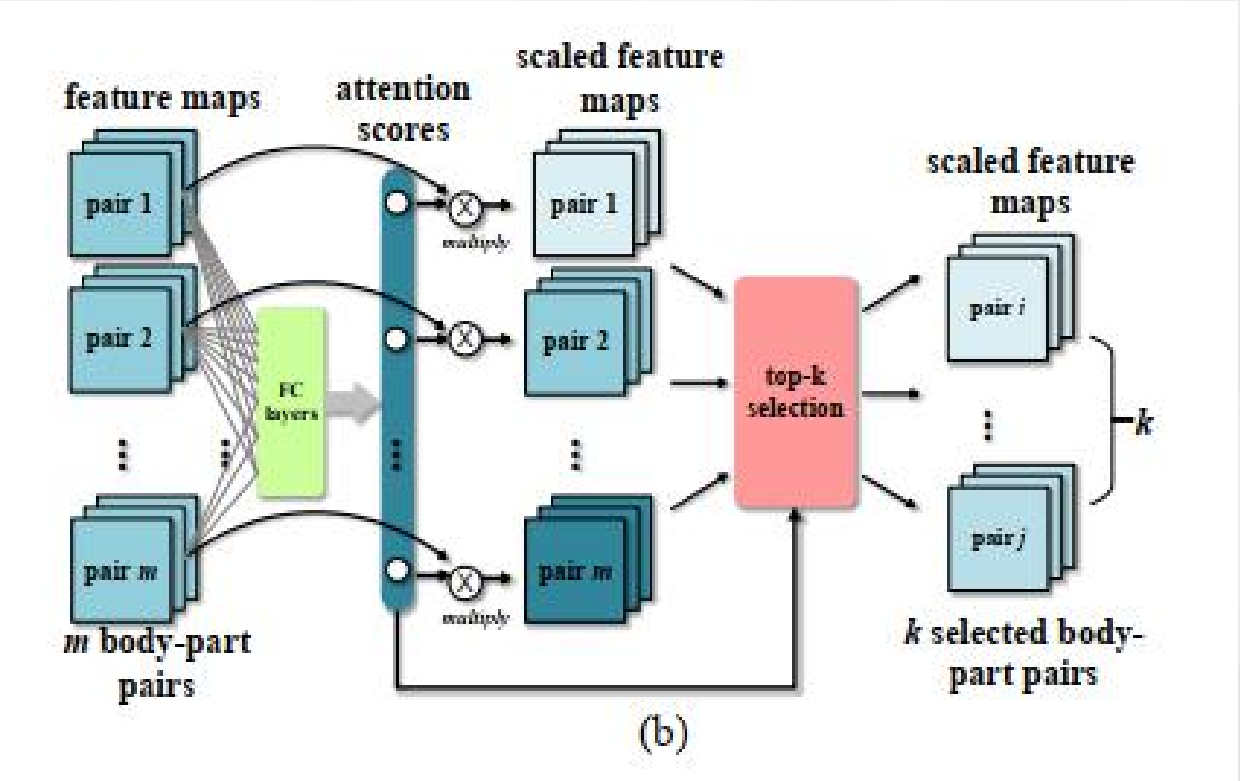
Pairwise Body-Part Attention for Recognizing Human-Object Interactions ——ROI-pairwise Pooling



Their idea is simple but novel. The value of the part of the union box that does not belong to the two bounding boxes is set to 0, and the features in the two bounding boxes remain original values. Then do ROI pooling on this new feature map. The author uses max pooling.

Pairwise Body-Part Attention for Recognizing Human-Object Interactions ——Attention Module

Here, the attention module is a self-attention structure, which uses its features to generate its attention. It means that the pooling feature passes through several fully connected layers to get a scalar as the weight. There are a total of m pairwise body parts, and m attention weights S are generated. Then, sorted according to the size of the weights, the first k pairwise body parts with the highest weights are selected and retained, and the rest are discarded. Finally, the attention weight is multiplied by the feature to obtain k pairwise body part features. These features are used for fusion with global features to improve the accuracy of HOI recognition.



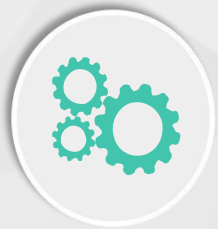
Pairwise Body-Part Attention for Recognizing Human-Object Interactions ——Experiments

Method	Full Im.	Bbox/Pose	MIL	Wtd Loss	mAP
AlexNet+SVM [5]	✓				19.4
R*CNN [14]		✓	✓		28.5
Mallya & Lazebnik [29]	✓	✓	✓		33.8
Pose Regu. Attn. Pooling [10]	✓	✓			34.6
Ours	✓	✓	✓		**37.5**
Mallya & Lazebnik, weighted loss [29]	✓	✓	✓	✓	36.1
Ours, weighted loss	✓	✓	✓	✓	**39.9**

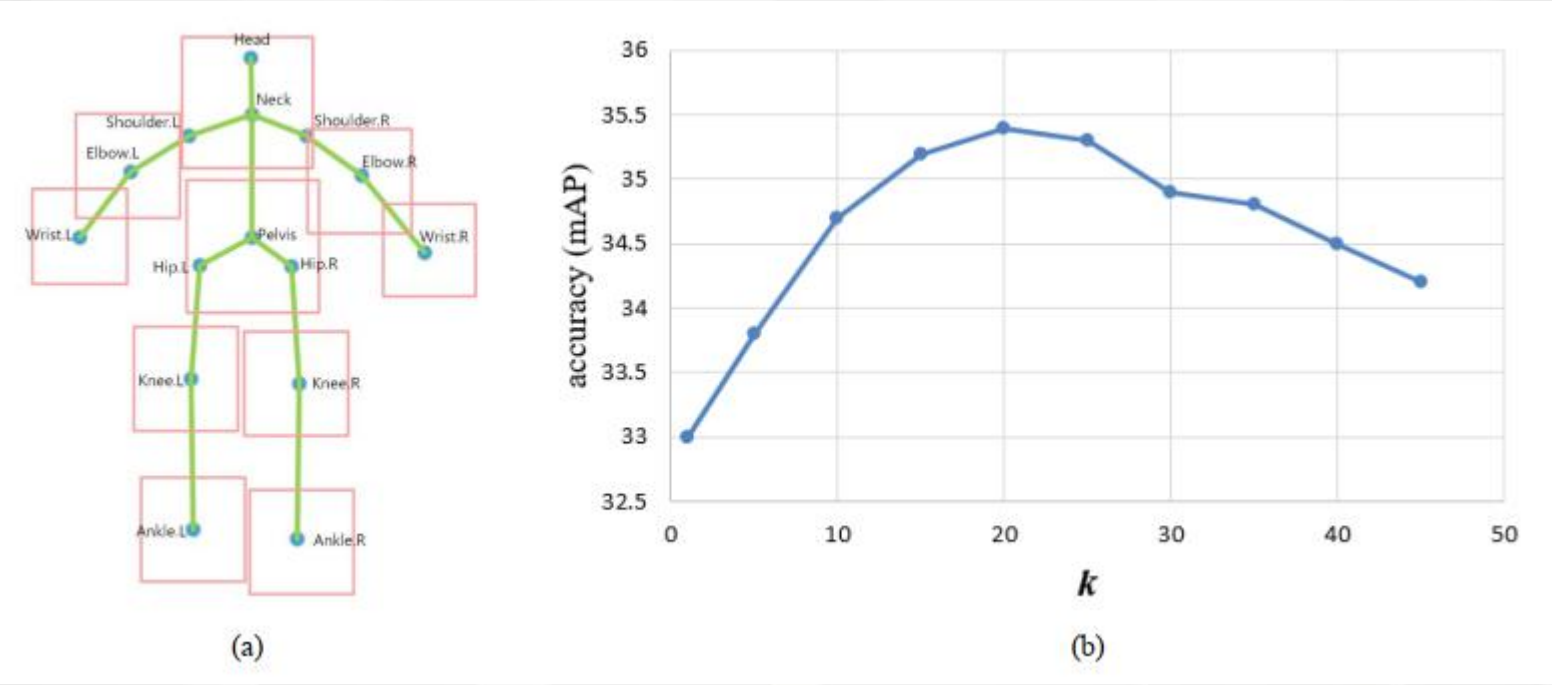
Table 1. Comparison with previous results on the HICO test set. The result of R*CNN is directly copied from [29].

Method	Full Img	Bbox	Pose	Val (mAP)	Test (mAP)
Dense Trajectory + Pose [2]	✓		✓	-	5.5
R*CNN, VGG16 [14]		✓		21.7	26.7
Mallya & Lazebnik, VGG16 [29]	✓	✓		-	32.2
Ours, VGG16	✓	✓	✓	**30.9**	**36.8**
Pose Reg. Attn. Pooling, Res101 [10]	✓		✓	30.6	36.1
Ours, Res101	✓	✓	✓	**32.0**	**37.5**

Table 2. Comparison with previous results on the MPII test set. The results on test set are obtained by e-mailing our predictions to the author of [2]



Pairwise Body-Part Attention for Recognizing Human-Object Interactions ——Experiments

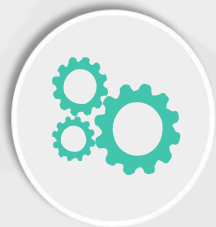


The possible reason is that too many pairwise body parts introduce a lot of useless features and noise, which is not conducive to the model's recognition of HOI.

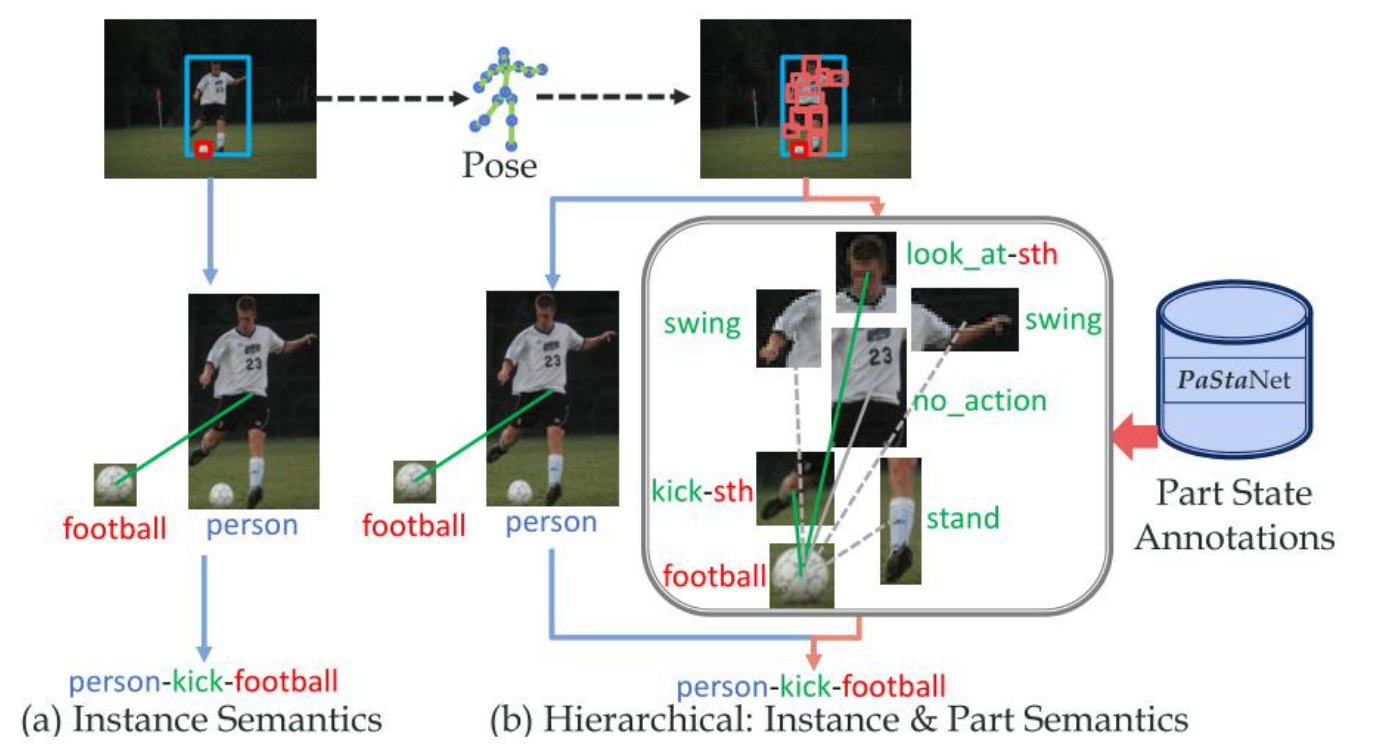
*PaSta*Net: Toward Human Activity Knowledge Engine

Yong-Lu Li, Liang Xu, Xinpeng Liu, Xijie Huang, Yue Xu,
Shiyi Wang, Hao-Shu Fang, Ze Ma, Mingyang Chen, Cewu Lu*
Shanghai Jiao Tong University
{yonglu_li, liangxu, otaku_huang, silicxuyue, shiywang}@sjtu.edu.cn,
{xinpengliu0907, fhaoshu}@gmail.com, {maze1234556, cmy_123, lucewu}@sjtu.edu.cn

An image-level behavior understanding method is proposed based on a knowledge engine (identifying different parts of the human body and the state of each part and inferring behavior information).

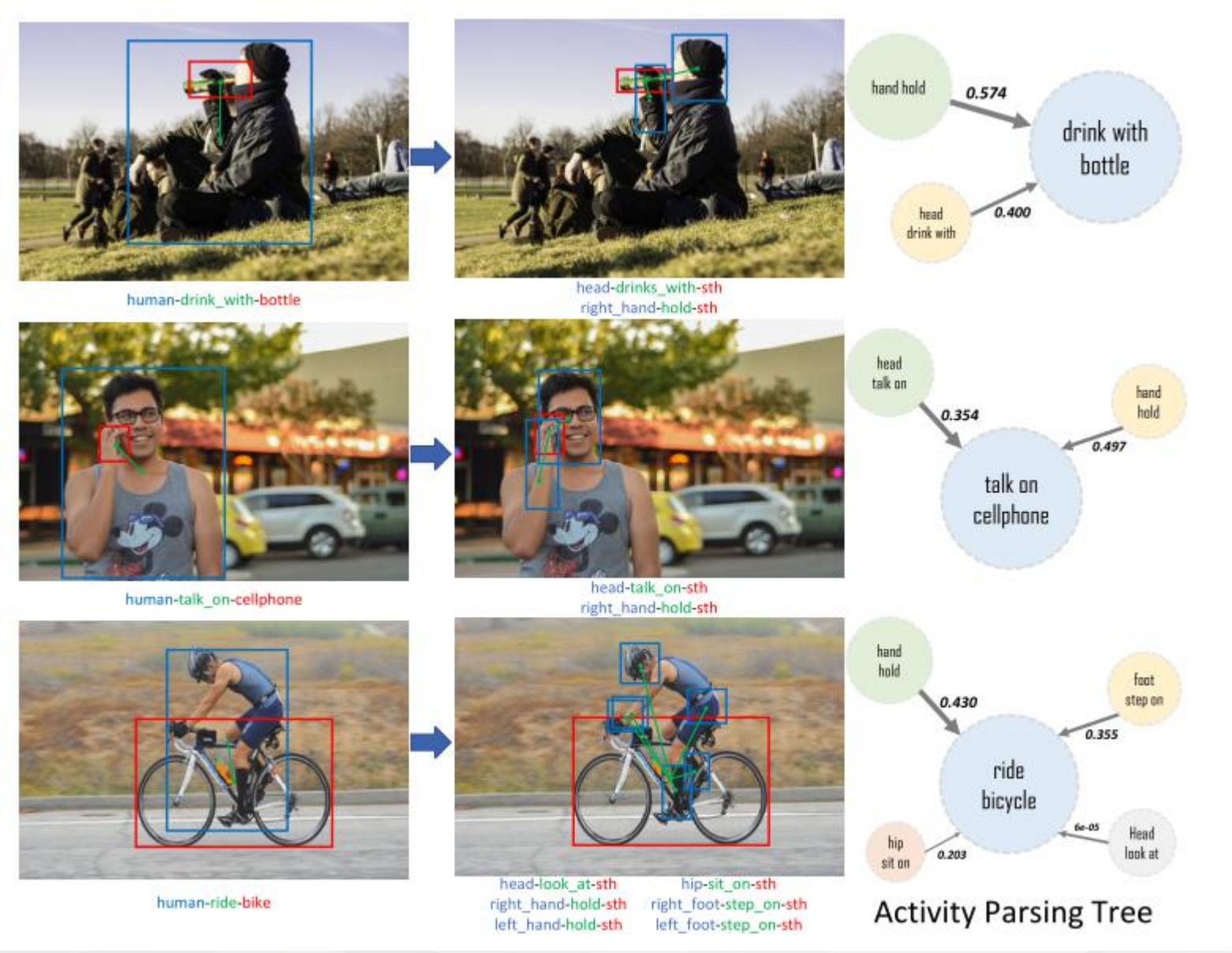


PaStaNet: Toward Human Activity Knowledge Engine



The previous methods are mostly based on instance-level features or knowledge (human, object) to learn actions. But for complex behaviors such as human-object interaction, this kind of coarse-grained knowledge is not enough, and the trained model is difficult to generalize (domain gap) due to different action types of different datasets. The overall idea of Human Activity Understanding is first to detect the various parts of the human and the corresponding state and then identify the behavior category through these states.

PaStaNet: Toward Human Activity Knowledge Engine——dataset



They divided people into ten parts: head, two upper arms, two hands, hip, two thighs, two feet. The number of states in each part is limited. They finally chose 118,000 pictures and 156 types of HOI tags. At the same time, they also marked body part states, which are the states of each part of the human body. There are about 220,000 labeling results through crowdsourcing labeling, as shown in the figure.

PaStaNet: Toward Human Activity Knowledge Engine——model

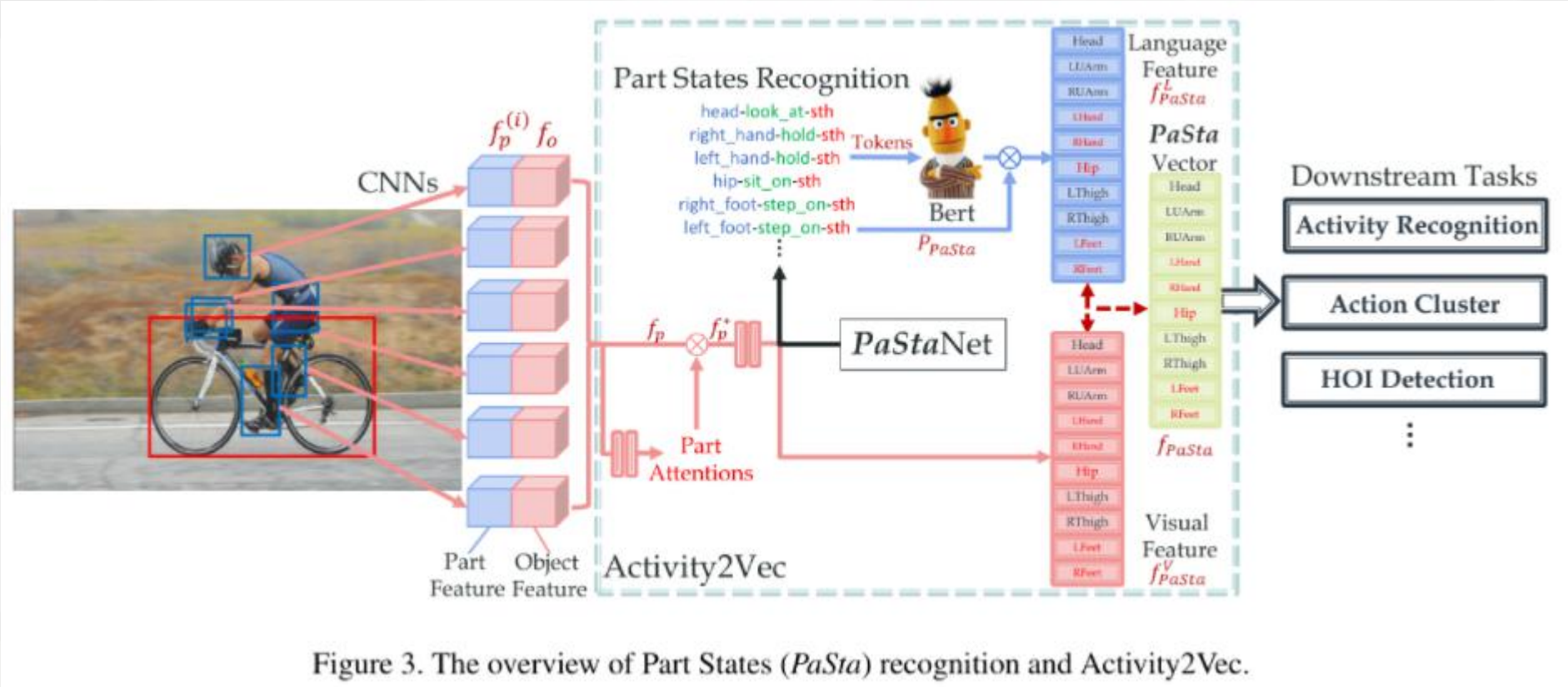


Figure 3. The overview of Part States (*PaSta*) recognition and Activity2Vec.

The structure of the model can be decomposed into the following parts: (1) Take images as input to obtain human parts; (2) Take images and human parts as input to obtain human parts states; (3) Take images, human parts and corresponding states are used as input to get the behavior category.

PaStaNet: Toward Human Activity Knowledge Engine——model

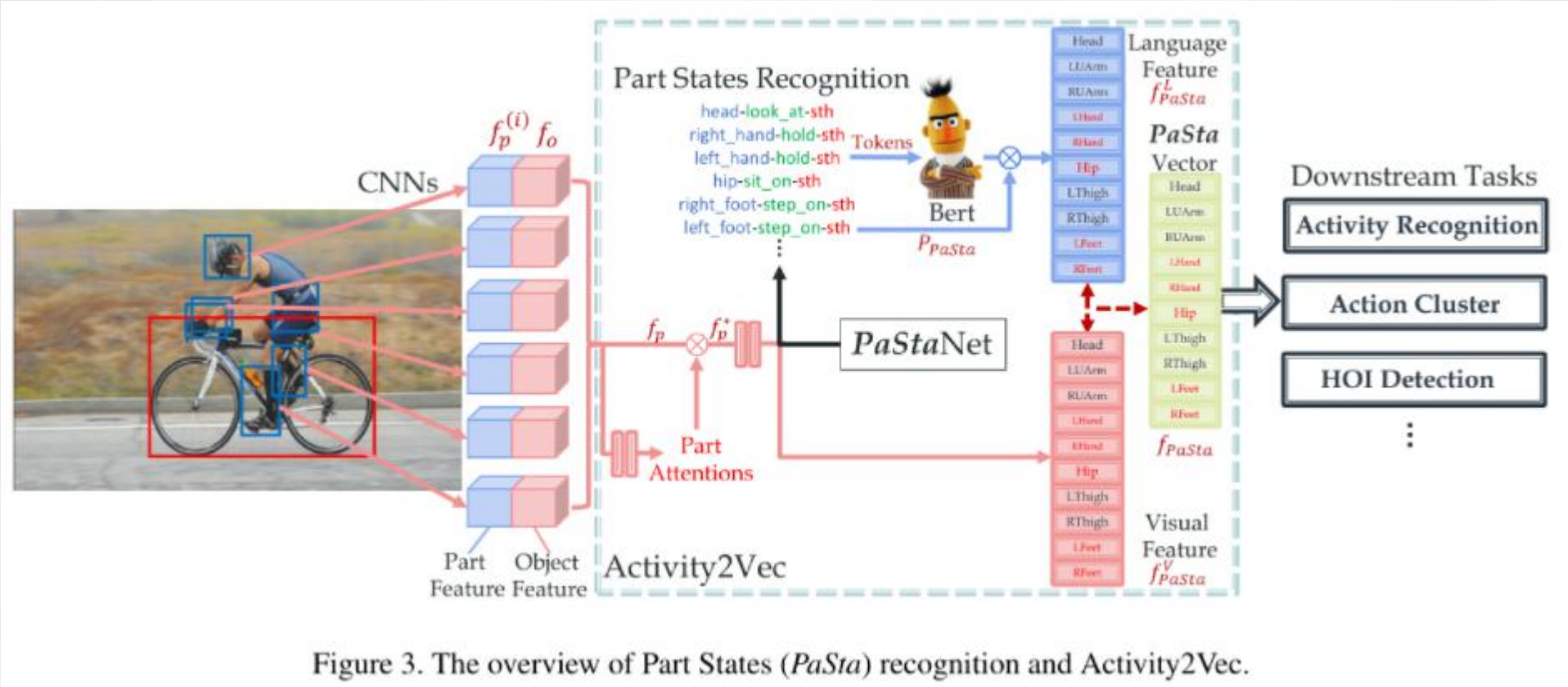


Figure 3. The overview of Part States (*PaSta*) recognition and Activity2Vec.

An attention alpha needs to be calculated first, which indicates the degree of relevance of a certain part's feature to the result, with a value of 0-1. They take the part and object features as input and calculate alpha through a part attention predictor. In this step, a loss function for the classification result of alpha is obtained. If the human part is related to the action of the human, then the ground truth label is 1, and if there is no connection, it is 0.

PaStaNet: Toward Human Activity Knowledge Engine——model

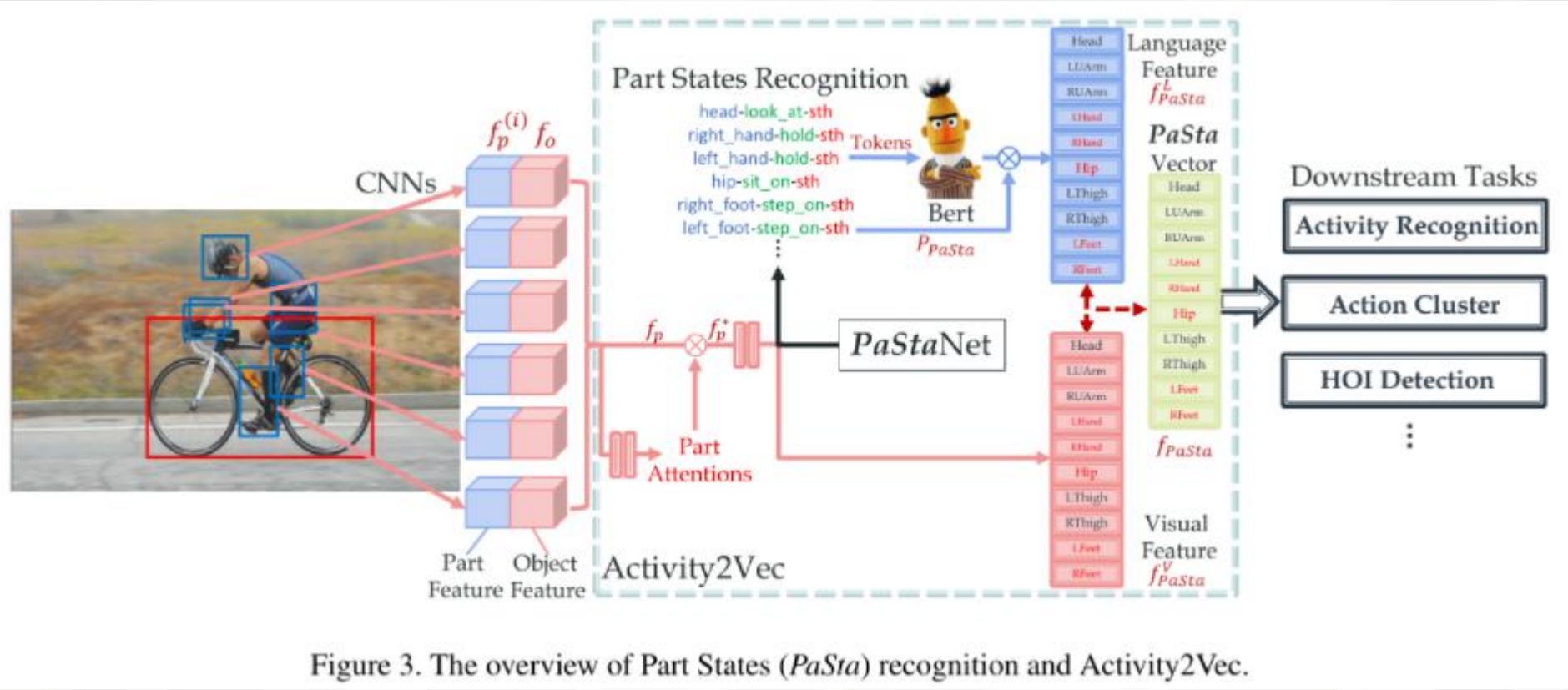


Figure 3. The overview of Part States (*PaSta*) recognition and Activity2Vec.

Then, the model needs to recognize PaSta(Part State). This is a multi-classification task; a human part may correspond to multiple states. The input of PaSta is human body features and object features, and the two features are operation by concatenation, max-pooling, and an FC layer to get the prediction result. In this step, the human body features are multiplied by the corresponding alpha calculated in the previous step.

PaStaNet: Toward Human Activity Knowledge Engine——model

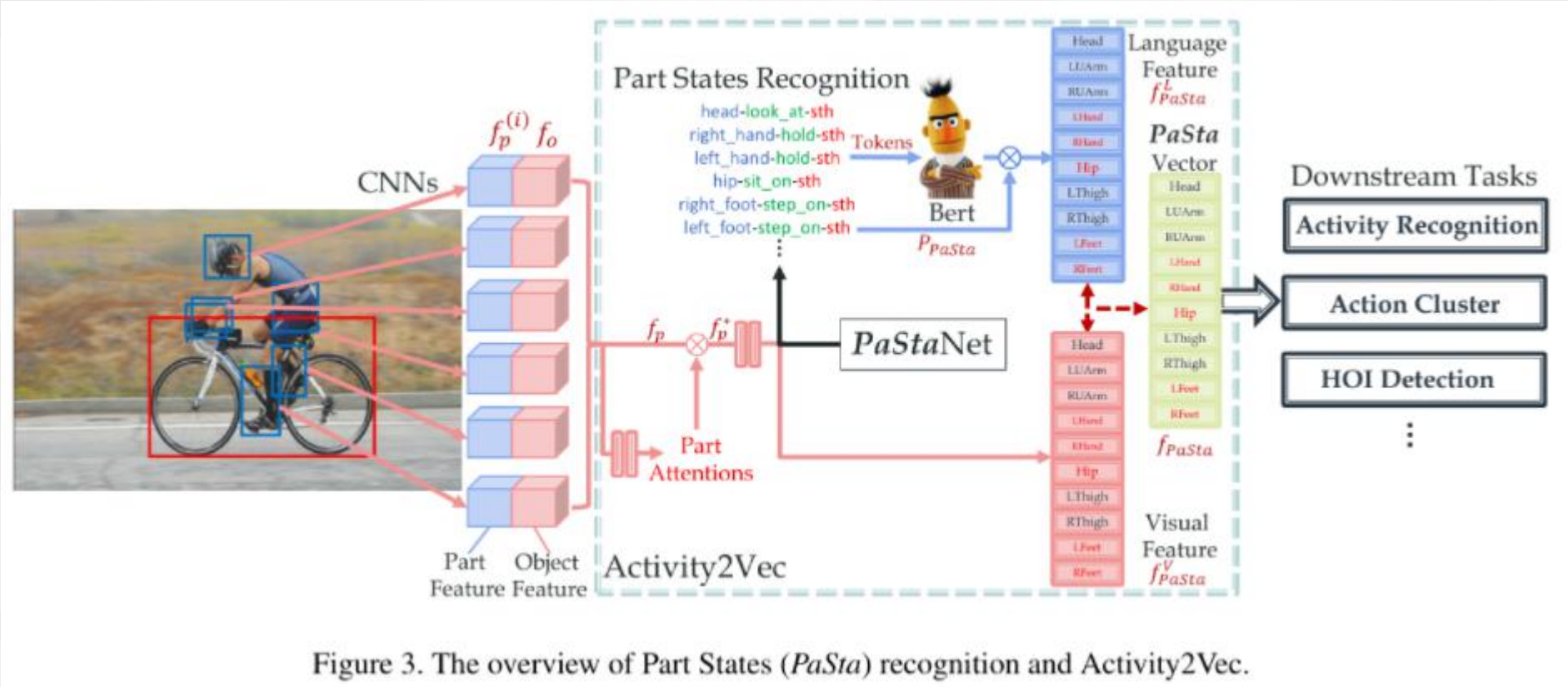
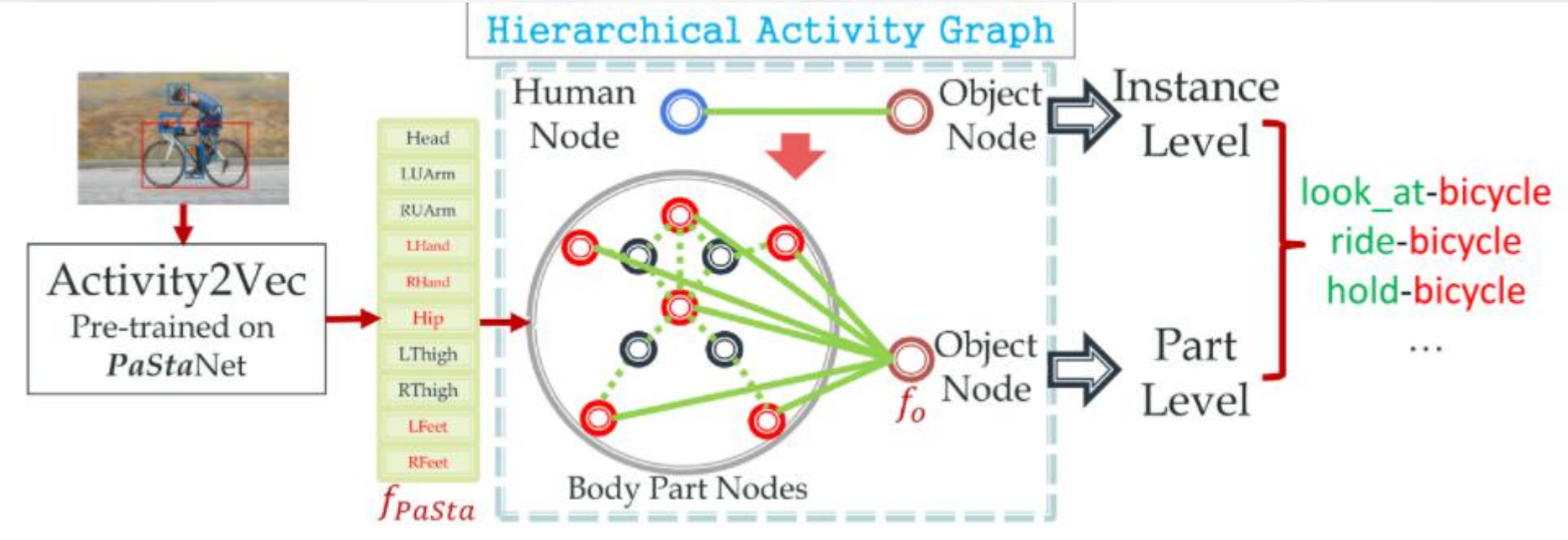


Figure 3. The overview of Part States (*PaSta*) recognition and Activity2Vec.

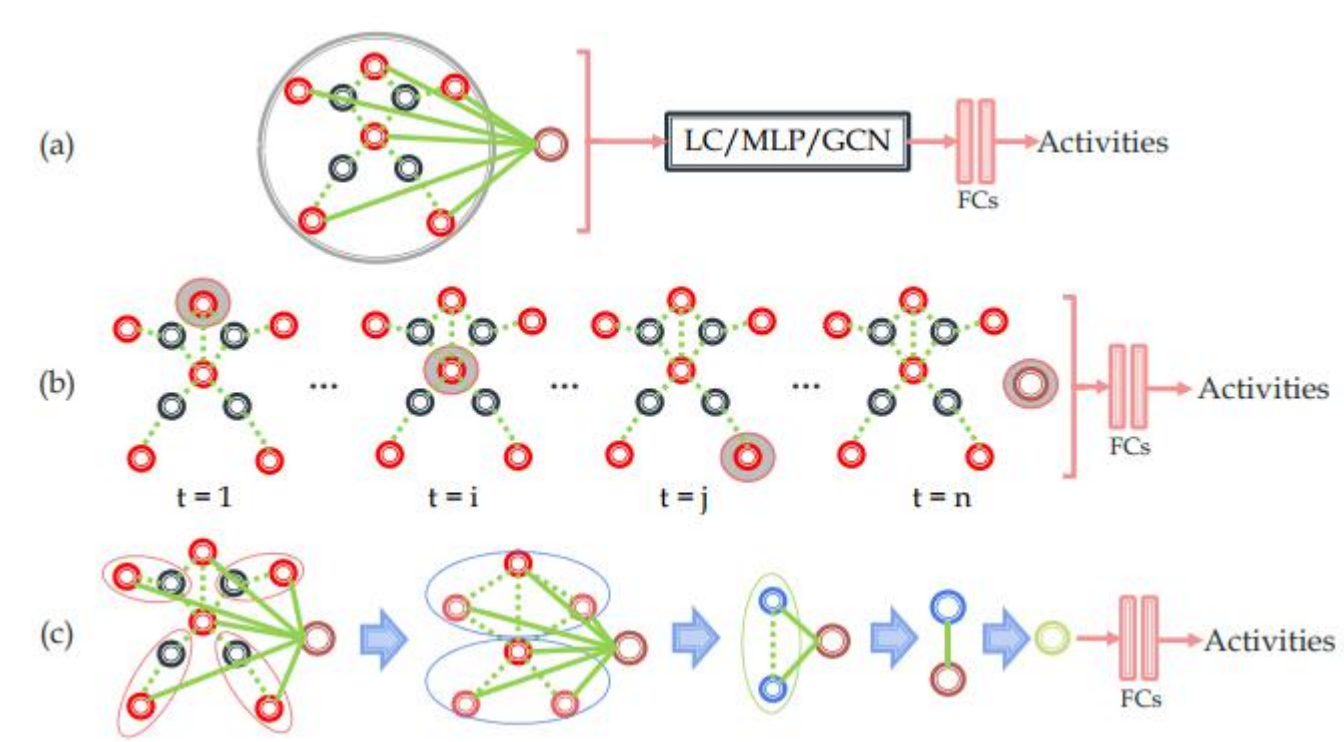
The last part converts the previous PaSta information into features for subsequent applications. It includes visual features and language features. The visual feature is a 512-dimensional vector obtained using the last FC of the previous PaSta Recognition. The language feature introduces a Bert and inputs these Pasta labels as tokens to obtain its new representation.

PaStaNet: Toward Human Activity Knowledge Engine——*PaSta-based Activity Reasoning*



Through Activity2Vec, the model obtains the two features of each part and then uses them for subsequent tasks. The overall idea is to use the instance-label feature and the part-label feature for post-fusion, but there are many specific ways.

PaStaNet: Toward Human Activity Knowledge Engine——*PaSta-based Activity Reasoning*



The article only briefly describes a few of them, including Linear combination, MLP, Graph Convolution network, sequential model, tree-structured passing, etc.

PaStaNet: Toward Human Activity Knowledge Engine——Experiments

Method	mAP	Few@1	Few@5	Few@10
R*CNN [21]	28.5	-	-	-
Girdhar *et al.* [18]	34.6	-	-	-
Mallya *et al.* [41]	36.1	-	-	-
Pairwise [14]	39.9	13.0	19.8	22.3
Mallya *et al.* [41]+*PaSta*Net*-Linear	45.0	26.5	29.1	30.3
Pairwise [14]+*PaSta*Net*-Linear	**45.9**	26.2	30.6	31.8
Pairwise [14]+*PaSta*Net*-MLP	45.6	26.0	**30.8**	**31.9**
Pairwise [14]+*PaSta*Net*-GCN	45.6	25.2	30.0	31.4
Pairwise [14]+*PaSta*Net*-Seq	**45.9**	25.3	30.2	31.6
Pairwise [14]+*PaSta*Net*-Tree	45.8	24.9	30.3	31.8
*PaSta*Net*-Linear	44.5	**26.9**	30.0	30.7
Pairwise [14]+GT-*PaSta*Net*-Linear	65.6	47.5	55.4	56.6
Pairwise [14]+*PaSta*Net-Linear	**46.3**	24.7	**31.8**	**33.1**

Table 1. Results on HICO. "Pairwise [14]+*PaSta*Net" means the late fusion of [14] and our part-level result. Few@i indicates the mAP on few-shot sets. @i means the number of training images is less than or equal to i. The HOI categories number of Few@1, 5, 10 are 49, 125 and 163. "*PaSta*Net-x" means different *PaSta-R*.

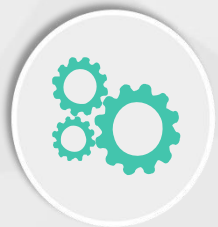
Method	Default			Known Object		
	Full	Rare	Non-Rare	Full	Rare	Non-Rare
InteractNet [22]	9.94	7.16	10.77	-	-	-
GPNN [48]	13.11	9.34	14.23	-	-	-
iCAN [17]	14.84	10.45	16.15	16.26	11.33	17.73
TIN [33]	17.03	13.42	18.11	19.17	15.51	20.26
iCAN [17]+*PaSta*Net*-Linear	19.61	17.29	20.30	22.10	20.46	22.59
TIN [33]+*PaSta*Net*-Linear	**22.12**	**20.19**	**22.69**	**24.06**	**22.19**	**24.62**
TIN [33]+*PaSta*Net*-MLP	21.59	18.97	22.37	23.84	21.66	24.49
TIN [33]+*PaSta*Net*-GCN	21.73	19.55	22.38	23.95	22.14	24.49
TIN [33]+*PaSta*Net*-Seq	21.64	19.10	22.40	23.82	21.65	24.47
TIN [33]+*PaSta*Net*-Tree	21.36	18.83	22.11	23.68	21.75	24.25
*PaSta*Net*-Linear	19.52	17.29	20.19	21.99	20.47	22.45
TIN [33]+GT-*PaSta*Net*-Linear	34.86	42.83	32.48	35.59	42.94	33.40
TIN [33]+*PaSta*Net-Linear	**22.65**	**21.17**	**23.09**	**24.53**	**23.00**	**24.99**

Table 2. Results on HICO-DET.

Method	$AP_{role}(Scenario1)$	$AP_{role}(Scenario2)$
Gupta *et al.* [25]	31.8	-
InteractNet [22]	40.0	-
GPNN [48]	44.0	-
iCAN [17]	45.3	52.4
TIN [33]	47.8	54.2
iCAN [17]+*PaSta*Net-Linear	49.2	55.6
TIN [33]+*PaSta*Net-Linear	**51.0**	**57.5**

Table 3. Transfer learning results on V-COCO [25].

They use PaSta to perform transfer learning directly. Like the ImageNet training backbone, Activity2Vec trained by PaStaNet can transfer a large amount of part knowledge to new tasks. For example, they have achieved very good results on large-scale behavioral data: HICO (+6.4mAP), HICO-DET (+5.0mAP); even cross-modality has a 3.6mAP improvement in the video data set AVA.



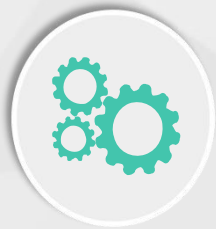
Masked Autoencoders Are Scalable Vision Learners

Kaiming He*,† Xinlei Chen* Saining Xie Yanghao Li Piotr Dollár Ross Girshick

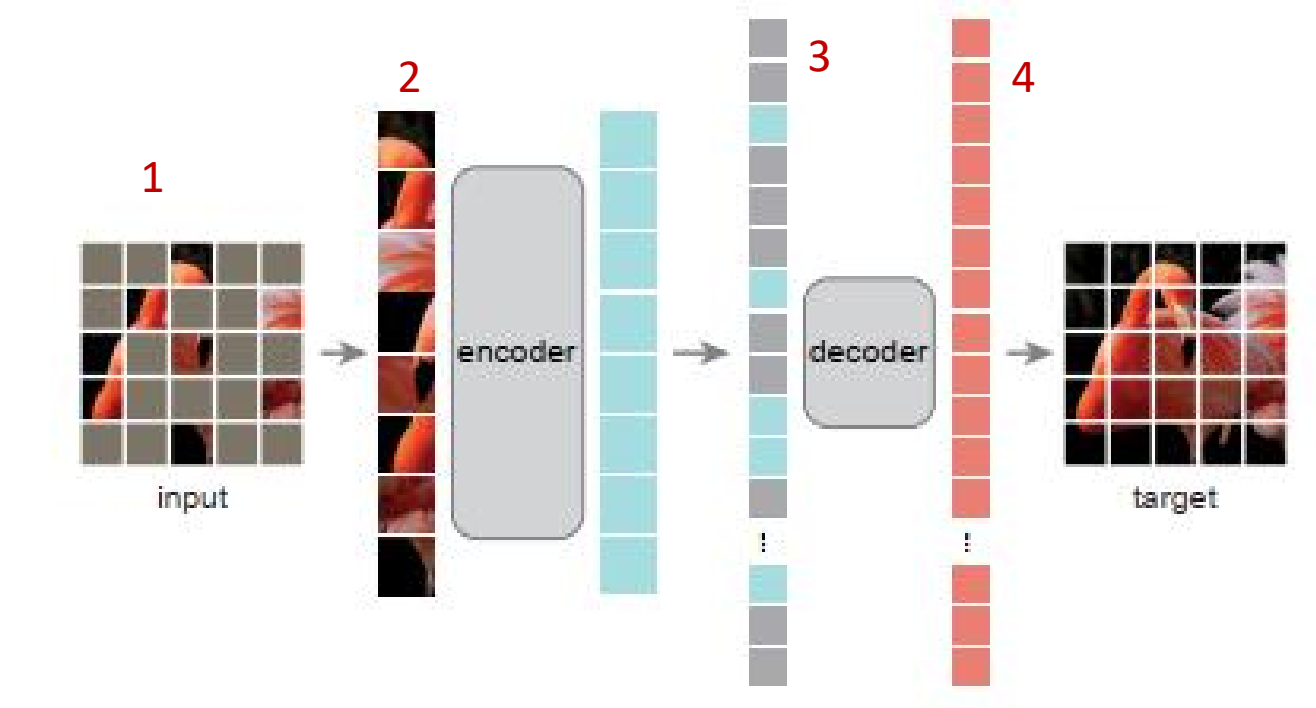
*equal technical contribution †project lead

Facebook AI Research (FAIR)

Its idea and structure are very simple.
It innovatively introduces the form like NLP's Bert into the CV field.

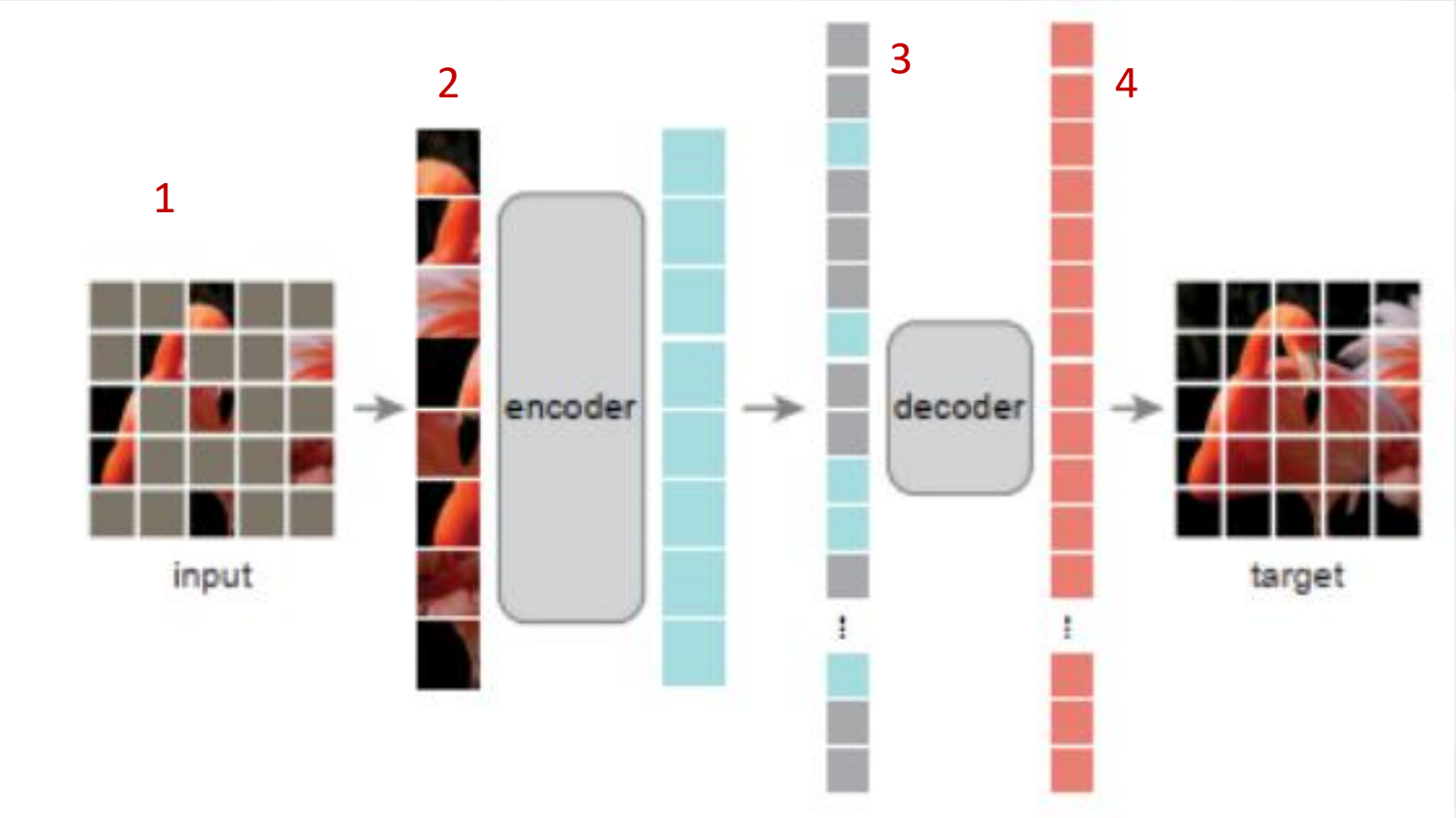


Masked Autoencoders Are Scalable Vision Learners——Structure



1. Divide the patch and mask out 75% of the patch simultaneously.
2. Then, input the visible patch into the encoder, using such a method of the Vit model.
3. The input is spliced with the mask patch and passed through a decoder.
4. After the decoder, a hidden linear layer maps it to the original image patch dimension for the loss of MSE.

Masked Autoencoders Are Scalable Vision Learners——How to utilize it

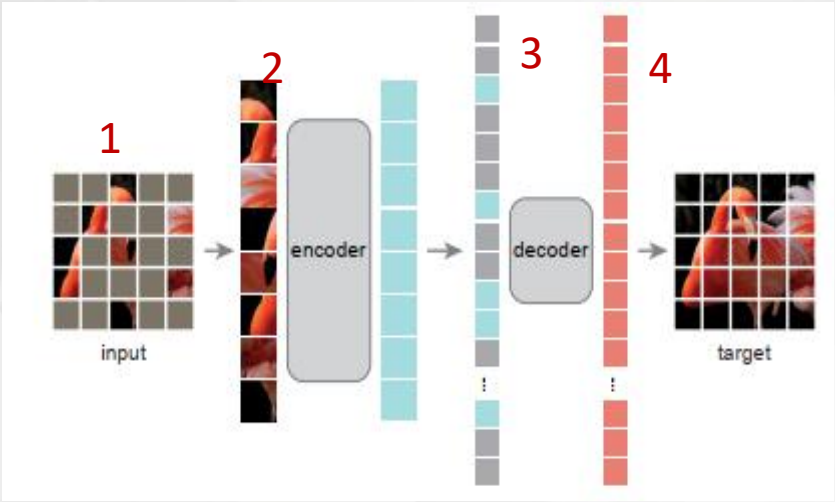


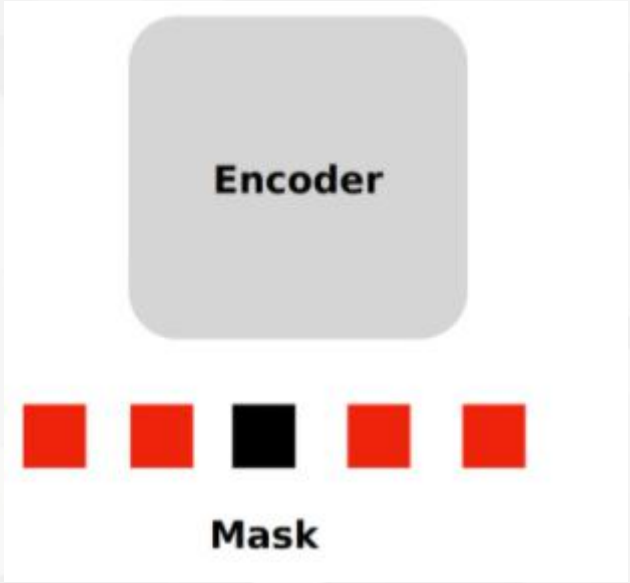
The encoder is extracting the features of the visible image

The decoder is reconstructing the image through the extracted image features

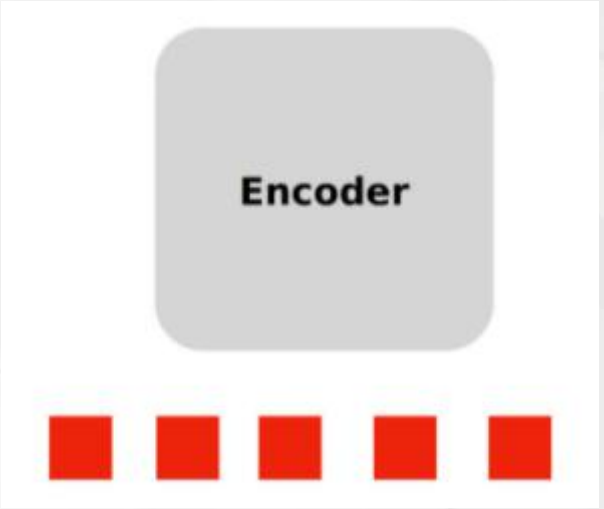
Masked Autoencoders Are Scalable Vision Learners——

Problem： Why is the masked part placed on the decoder side?





Pre-training

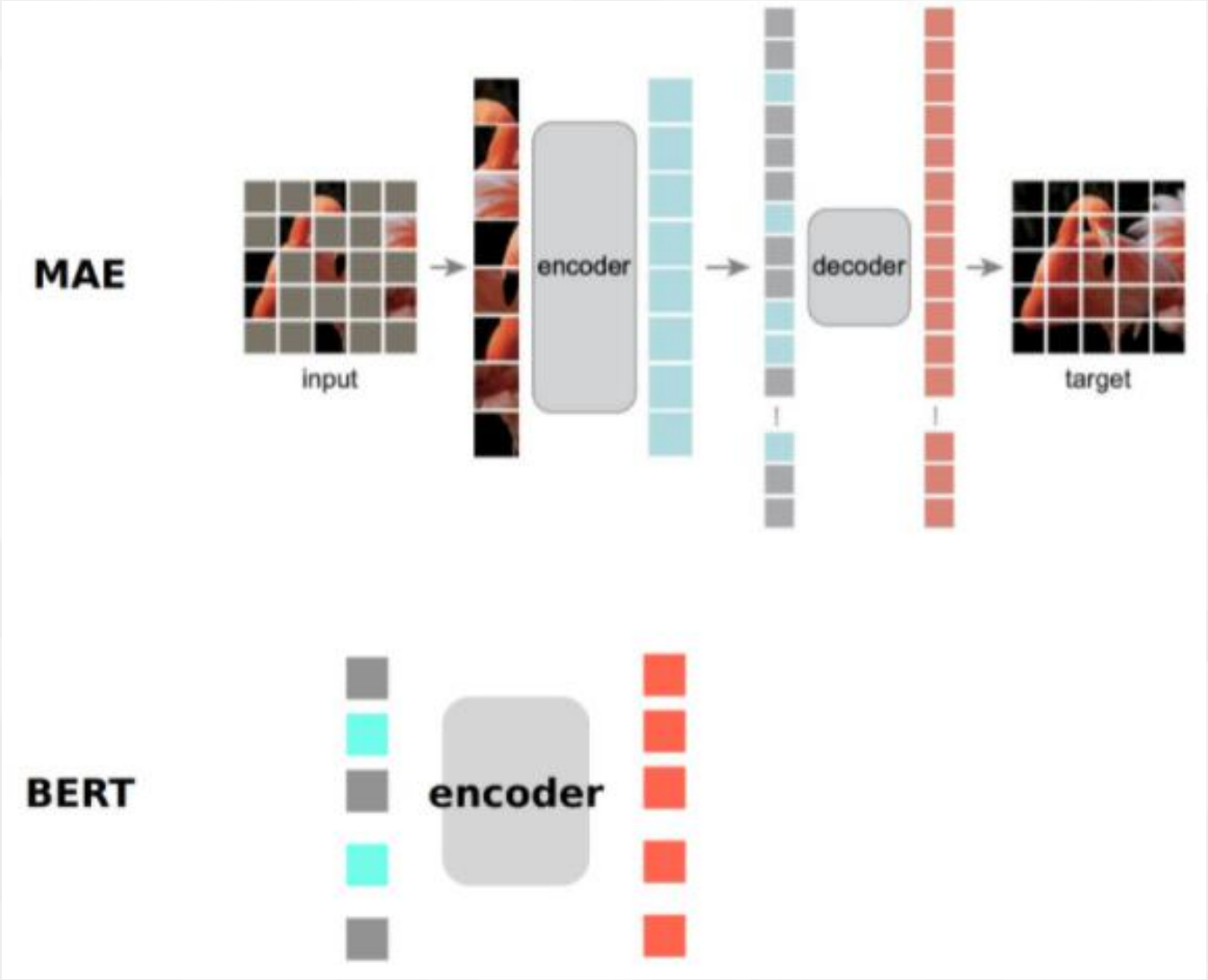


Fine-tuning for subsequent tasks

In the pre-training, the mask features are input together with the original features and the loss is calculated, but in the fine-tuning process for subsequent tasks, the input is without mask features. Then there will be a gap between the two forms. So Bert will have a ratio of 811 during training, and only eight in ten will be truly masked. This is to reduce this gap. And MAE is directly trying to eliminate this gap.

Masked Autoencoders Are Scalable Vision Learners——

Problem： Why is the masked part placed on the decoder side?



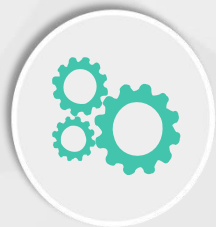
Masked Autoencoders Are Scalable Vision Learners——

Problem： Why is the masked part placed on the decoder side?

Masked Autoencoders Are Scalable Vision Learners——Summarize

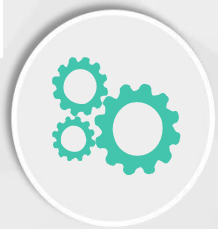
From Bert in NLP to MAE in CV

1. The basic architecture is different. The CNN was commonly used in cv before, so it isn't easy to introduce information such as location symbols and mask symbols. (The ViT model has solved this problem)
2. The information density is different. The text is a high-level information density, and the image is a low-level information density. Therefore, bert needs to mask 15% of the information for the text, while MAE needs to mask 75% of the information. Because the mask is only 15%, the model can be lazy, and it can also create such low-level information density pixels to restore the image. Therefore, the model can learn high-level semantic information after increasing the difficulty.
3. Self-encoding decoder. Because of the different information density, cv cannot directly use Bert's decoder; it must design a decoder for cv.



Masked Autoencoders Are Scalable Vision Learners——Visualization Experiment Results

Figure 2. Example results on ImageNet *validation* images. For each triplet, we show the masked image (left), our MAE reconstruction† (middle), and the ground-truth (right). The masking ratio is 80%, leaving only 39 out of 196 patches. More examples are in the appendix.
†*As no loss is computed on visible patches, the model output on visible patches is qualitatively worse. One can simply overlay the output with the visible patches to improve visual quality. We intentionally opt not to do this, so we can more comprehensively demonstrate the method's behavior.*



Masked Autoencoders Are Scalable Vision Learners——Visualization Experiment for Versatility

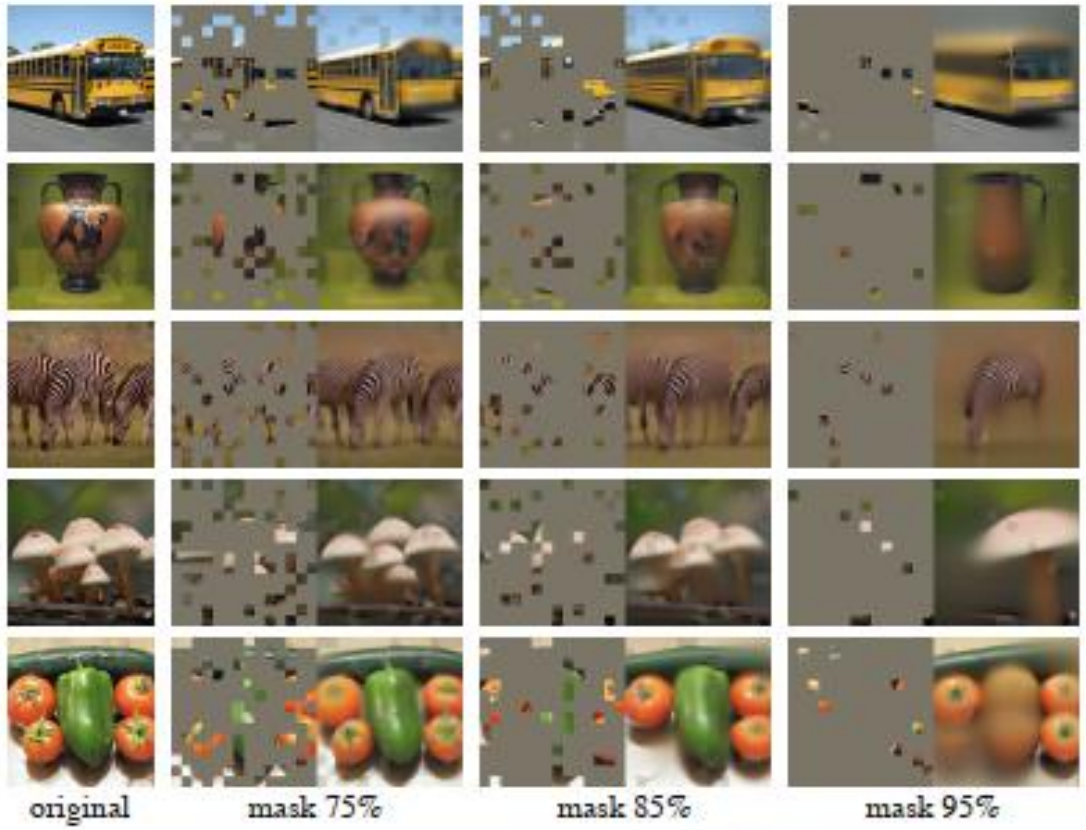


Figure 4. Reconstructions of ImageNet *validation* images using an MAE pre-trained with a masking ratio of 75% but applied on inputs with higher masking ratios. The predictions differ plausibly from the original images, showing that the method can generalize.

Masked Autoencoders Are Scalable Vision Learners——Experiment for Different Measurement Methods

Two measurement methods:

1. Linear Probe: Fix the parameters of the encoder and only learn the parameters of the linear layer before the classification task.
2. Fine Tune: The entire model includes the encoder to learn together.

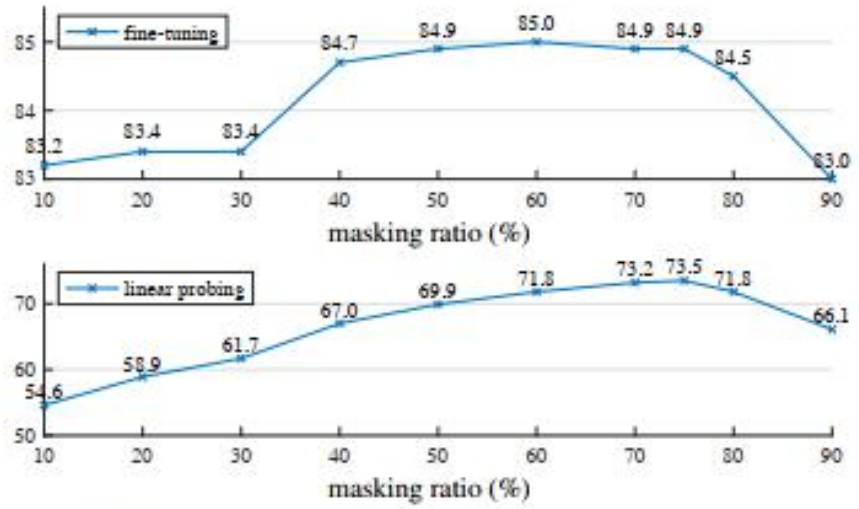


Figure 5. **Masking ratio**. A high masking ratio (75%) works well for both fine-tuning (top) and linear probing (bottom). The y-axes are ImageNet-1K validation accuracy (%) in all plots in this paper.

blocks	ft	lin
1	84.8	65.5
2	**84.9**	70.0
4	**84.9**	71.9
8	**84.9**	**73.5**
12	84.4	73.3

(a) **Decoder depth**. A deep decoder can improve linear probing accuracy.

dim	ft	lin
128	**84.9**	69.1
256	84.8	71.3
512	**84.9**	**73.5**
768	84.4	73.1
1024	84.3	73.1

(b) **Decoder width**. The decoder can be narrower than the encoder (1024-d).

case	ft	lin	FLOPs
encoder w/ [M]	84.2	59.6	3.3×
encoder w/o [M]	**84.9**	**73.5**	**1**×

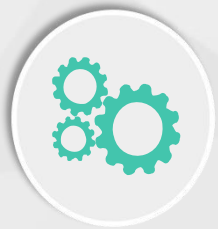
(c) **Mask token**. An encoder without mask tokens is more accurate and faster (Table 2).

Masked Autoencoders Are Scalable Vision Learners——Experiment for Different Mask Types

Figure 6. **Mask sampling strategies** determine the pretext task difficulty, influencing reconstruction quality and representations (Table 1f). Here each output is from an MAE trained with the specified masking strategy. Left: random sampling (our default). Middle: block-wise sampling [2] that removes large random blocks. Right: grid-wise sampling that keeps one of every four patches. Images are from the validation set.

case	ratio	ft	lin
random	75	**84.9**	**73.5**
block	50	83.9	72.3
block	75	82.8	63.9
grid	75	84.0	66.0

(f) **Mask sampling**. Random sampling works the best. See Figure 6 for visualizations.



Thank you