

Baidu-UTS Submission to the EPIC-Kitchens Action Recognition Challenge 2019

Xiaohan Wang^{1,2}, Yu Wu^{1,2}, Linchao Zhu², Yi Yang^{1,2}

{xiaohan.wang-3,yu.wu-3,linchao.zhu}@student.uts.edu.au; yi.yang@uts.edu.au

¹Baidu Research, ²The ReLER lab, CAI, University of Technology Sydney

Abstract

In this report, we present the Baidu-UTS¹ submission to the EPIC-Kitchens Action Recognition Challenge in CVPR 2019. This is the winning solution to this challenge. In this task, the goal is to predict verbs, nouns, and actions from the vocabulary for each video segment. The EPIC-Kitchens dataset contains various small objects, intense motion blur, and occlusions. It is challenging to locate and recognize the object that an actor interacts with. To address these problems, we utilize object detection features to guide the training of 3D Convolutional Neural Networks (CNN), which can significantly improve the accuracy of noun prediction. Specifically, we introduce a Gated Feature Aggregator module to learn from the clip feature and the object feature. This module can strengthen the interaction between the two kinds of activations and avoid gradient exploding. Experimental results demonstrate our approach outperforms other methods on both seen and unseen test set.

1. Introduction

Egocentric (first-person) video analysis is an important task but less explored than third-person video understanding. It is valuable for practical applications such as human-computer interaction, intelligent wearable devices, and service robots. Due to the lack of sufficiently large datasets, the progress in this area has been relatively slow. Recently, a large-scale egocentric video dataset named EPIC-Kitchens [5] has been released, which provides a new benchmark and has attracted much attention. The EPIC-Kitchens dataset is the largest dataset in first-person vision so far. It consists of 55 hours of recordings capturing all daily activities in the kitchens. The recognition task on the EPIC-Kitchens dataset is to predict the verb, noun, and the combination pair in each video segment.

*This work was done when Xiaohan Wang and Yu Wu were interned at Baidu Research. Part of this work was done when Yi Yang was visiting Baidu Research during his Professional Experience Program.

¹This submission is a joint work by the ReLER lab at UTS and Baidu Research.

Egocentric action recognition is a challenging task. Compared to third-person activity recognition, it requires to distinguish the object that human is interacting with from various small objects. The intense camera motion and occlusion make it more difficult to obtain an accurate prediction. Therefore, direct adoption of algorithms like 3D CNN that work for third-person video recognition may not achieve promising results on this task.

To address this problem, we introduce an object detection model to extract more precise object-related features to guide the training of 3D CNN. Specifically, we extract the clip feature and the object feature using 3D CNN and Faster R-CNN [14], respectively. Later, the two features are sent to a Gated Feature Aggregator module to produce a new representation for the final classification. This module can stabilize the training process and strengthen the interaction of the two different activations. To make the object feature more robust to the motion blur and occlusion, we feed the context frames between the center of the video clip to the detection model. Our method outperforms the baseline models and achieves the state-of-the-art on the test sets.

2. Related Work

Third-person video classification has attracted lots of research works in the last a few years. Two-stream convolutional networks [15] utilize optical flow information for motion modeling, while 3D convolutional networks [17, 4, 20, 22] recently achieved better performance than its 2D counterpart. Recurrent Neural Networks (RNNs) are effective architectures for long sequence modeling and have been found useful for video classification in [23, 1]. Other aggregation methods like VLAD [21], actionVLAD [8] are also commonly used.

We discuss several methods evaluated on the EPIC-Kitchens dataset. The authors of EPIC-Kitchens provide a baseline result on the recognition benchmark. They train Temporal Segment Network (TSN) [18] to predict both verb and noun classes jointly. The two-stream TSN achieves the best performance on verb prediction and RGB-TSN outperforms their other models for noun prediction. However, without special design for egocentric videos, this state-of-

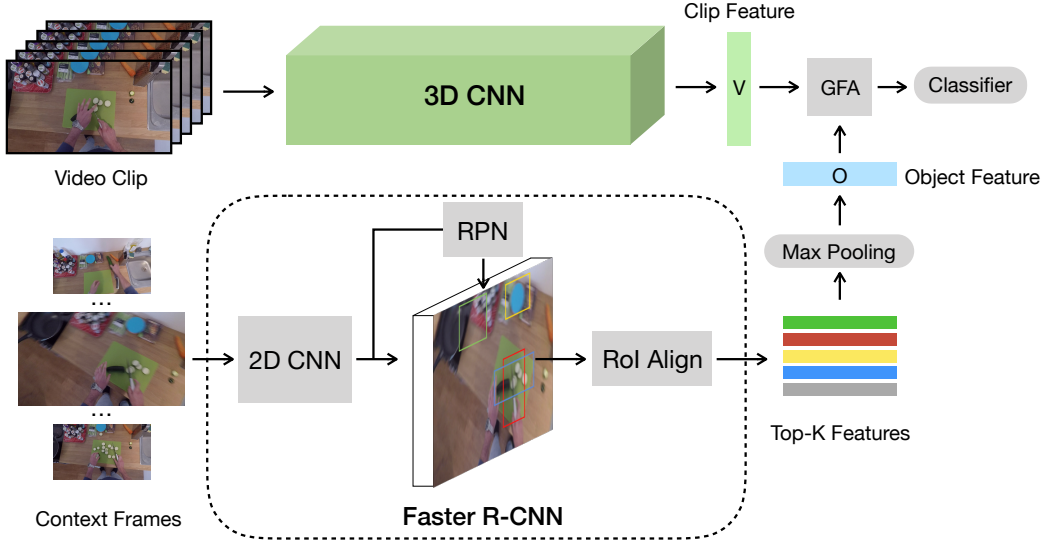


Figure 1. The overall framework of our approach.

the-art method for third-person video recognition does not achieve promising results, especially on noun classification.

The attention mechanism is efficient to locate the region of interest on the feature map. Sudhakaran et al. [16] propose a Long Short-Term Attention model to focus on features from relevant spatial parts. They extend LSTM with a recurrent attention component and an output pooling component to track the discriminative area smoothly across the video sequence. Their model obtains a significant gain over the TSN baseline.

The object detection model is another powerful way to extract object-related features. Baradel et al. [3] propose to perform object-level visual reasoning about spatio-temporal interactions in videos through the integration of object detection networks. More recently, Wu et al. [19] combine Long-Term Feature Banks that contains object-centric detection features with 3D CNN to improve the accuracy of noun recognition.

According to the success of image recognition, pretraining on large scale dataset can boost the performance of deep learning models. Ghadiyaram et al. [7] construct a large-scale video dataset with verb-noun label space. They pretrain a deep 3D CNN on the data and then finetune the model on EPIC-Kitchens. Their model achieves relatively high results, especially on the unseen test set.

3. Our Approach

As shown in Fig. 1, our framework consists of two branches. The first 3D CNN branch takes the sampled video clip as input and produces a clip feature. The second branch aims to extract the object-related features from the context frames. We sampled the frames within a window size w

at the center of the current clip. Then the pretrained object detector processes them frame by frame. We choose the top-K bounding boxes with the highest score and use RoIAlign [10] to get the features from the feature maps of the 2D CNN. After that, the top-K features are max pooled and send to the Gated Feature Aggregator (GFA) module with the clip feature. This module can guide the model to utilize the object-related information and find more discriminative channels. We describe the details of GFA in Sec. 3.2. The output of GFA is our final feature and is used to classify verbs and nouns.

3.1. Base Models

We use three 3D CNN backbones to extract video clip features. The first one is I3D [4] which is proposed by Carreira and Zisserman. They inflate 2D CNN architectures to 3D and initialize the network with ImageNet [6] pretrained weights. We use the two-stream I3D for verb classification and RGB I3D for noun classification. We do not use optical flow inputs for noun as it does not contain enough information of object appearance. The other two backbones we used are 3D ResNet-50 [9] and 3D ResNeXt-101[9]. They have similar architectures as the 2D models, but the convolutional kernels are in 3D. All the three 3D CNN backbones are pretrained on the Kinetics-400 dataset [4].

We use Faster R-CNN [14] pipeline to detect objects and extract object features. The backbone of the detector is 2D ResNeXt-101 [20] with FPN [12], which is trained on 1600-class Visual Genome [11, 2] and then finetuned on EPIC-Kitchens [5] detection dataset. We train two detectors following the above steps. One is $32 \times 8d$ ResNeXt-101 with 1024-dim output, and the other is $64 \times 4d$ ResNeXt-101 with 2048-dim output.

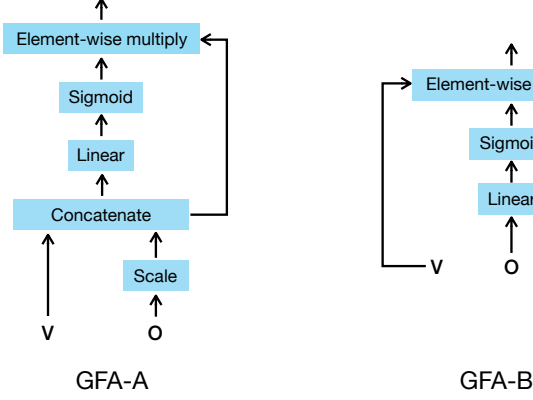


Figure 2. The two different types of GFA.

3.2. Gated Feature Aggregator

Wu et al. [19] propose to concatenate the object feature and the clip feature directly as the final representation. However, in our experiments, this method is sensitive to the backbones of 3D CNN and detector. When the two branches have different backbones, e.g., I3D and ResNext, the training loss is difficult to converge thus the final performance is not improved. To stabilize the training process and leverage the interdependencies of these two features, we design a Gated Feature Aggregator (GFA) module. As illustrated in Fig.2, GFA has two types.

GFA-A. Since the amplitudes of the object feature o and the clip feature v might be different, we scale o to enforce its amplitude to be approximate with v . The scaling operation can be performed by dividing a scalar. Another way of scaling is to multiply the ℓ_2 -normed o by the amplitude of v . After that, the concatenated o and v is transformed to a new representation by self-gating mechanism [13]. Formally, the output feature is computed as follows,

$$F = \sigma(W[v, \text{scale}(o)] + b) \cdot [v, \text{scale}(o)], \quad (1)$$

where “[]” indicates the concatenation operation. We have two motivations behind this design. First, we wish to avoid the gradient explosion by scale operation. Second, we want to strengthen the object-related channel using the gating operation.

GFA-B. The instability of the training process is mainly caused by the concatenation operation. In this type, we multiply gated o by v in an element-wise manner instead of concatenation. The final representation F is obtained as follows,

$$F = \sigma(Wo + b) \cdot v. \quad (2)$$

3.3. Action Re-weighting

The actions are determined by the pairs of verb and noun. The basic method of obtaining the action score is to calcu-

3D CNN	Baseline	Concat	GFA
ResNet-50	25.07	29.74	32.99
I3D RGB	27.92	23.97	34.13

Table 1. Comparison of different models for noun recognition on the new train/val set (Top-1 accuracy).

3D CNN	Detector	GFA	top-1	top-5
ResNet-50	-	-	55.62	81.60
ResNeXt-101	-	-	57.43	81.46
I3D RGB	-	-	59.38	82.78
I3D Flow	-	-	56.65	80.79
I3D two-stream	-	-	61.44	83.60
ResNet-50	1024 dim	Type A	57.61	82.64
Fusion	-	-	63.15	84.57

Table 2. Performance of different models for verb recognition on the new train/val set.

3D CNN	Detector	GFA	top-1	top-5
ResNet-50	-	-	25.07	46.84
ResNeXt-101	-	-	25.68	46.52
I3D RGB	-	-	27.92	52.85
ResNet-50	1024 dim	Type A	31.79	56.80
ResNet-50	2048 dim	Type A	32.99	57.81
ResNeXt-101	1024 dim	Type A	30.79	56.69
I3D RGB	1024 dim	Type B	31.14	58.42
I3D RGB	2048 dim	Type B	34.13	60.36
Fusion	-	-	39.09	65.00

Table 3. Performance of different models for noun recognition on the new train/val set.

late the multiplication of verb probability and noun probability. However, there are thousands of combinations and most verb-noun pairs that do not exist in reality, e.g. “open the knife”. In fact, there are only 149 action classes that have more than 50 samples [5]. Following the approach in [19], we re-weight the final action probability by a prior, i.e.

$$P(\text{action} = (v, n)) = \mu(v, n)P(\text{verb} = v)P(\text{noun} = n), \quad (3)$$

where μ is the occurrence frequency of action in training set.

4. Experiments

In all experiments, the inputs for 3D CNN are 64-frame video clips. The clips are randomly scaled and cropped to 224×224 . We choose the top-10 bounding boxes of the context frames to extract object features. We adopt the stochastic gradient descent (SGD) with momentum 0.9 for model training.

We train our model for verb and noun independently. To validate our models, we split the training data to the new

Model	data split	re-weighting	verb		noun		action	
			top-1	top-5	top-1	top-5	top-1	top-5
fused model	train/val	w/o	63.15	84.57	39.09	65.00	27.68	48.07
fused model	train/val	w	63.15	84.57	39.09	65.00	28.98	49.78
fused model	trainval/test-s1	w	69.80	90.95	52.27	76.71	41.37	63.59
fused model	trainval/test-s2	w	59.68	82.69	34.14	62.38	25.06	45.95

Table 4. Performance of the fused model on the train/val and trainval/test set.

training and validation set following [3].

For verb recognition, as shown in Table 2, we train five different models on the new training set and evaluate their top-1 and top-5 accuracy on the validation set. The two-stream I3D model (late fusion of I3D RGB and I3D flow) obtains the best performance, which can achieve 61.44% top-1 accuracy and 83.60% top-5 accuracy. Our ResNet-50 with GFA improves the top-1 accuracy by 1.99% than the baseline model, where GFA is type A with the norm and scale operation.

For noun recognition, as shown in Table 3, we experiment with eight models on the new train/val set. Due to the large margin improvement of the performance of noun recognition, we try more combinations of 3D CNN, detectors, and GFA. ResNet-50 with 2048-dim detection features and GFA-A results in 7.92% improvement of top-1 accuracy. Besides, I3D RGB model with 2048-dim detection features and GFA-B achieves the highest top-1 accuracy at 34.13%. As shown in Table. 1, the GFA module is more efficient than the direct concatenation.

For action recognition, as shown in Table 4, we calculate the final action top-1 and top-5 accuracy of our fused model on train/val split in two ways. The re-weighting strategy improves the top-1 accuracy by 1.30% and top-5 accuracy by 1.71%.

For the final submission, we train the above models on the whole training data. Our model ensemble achieves the best performance on both seen (s1) and unseen (s2) test set. The final results are shown in Table 4.

5. Discussion and Future Work

In this paper, we report our method details for the EPIC-Kitchens action recognition task. We combine object features with the clip feature to predict the action. To stabilize the training process and strengthen the interaction of different activations, we introduce a Gated Feature Aggregator, which has been validated to be important for feature learning. Our model achieves the state-of-the-art on both seen and unseen test data.

Our model is simple and does not introduce any interaction between the verb branch and the noun branch during training time. It can also be combined with the multi-modal information such as the narration of each segment following video-language models [24]. It is also promising to adopt

the model on action anticipation task owing to the more precise object features.

References

- [1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016.
- [2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, 2018.
- [3] Fabien Baradel, Natalia Neverova, Christian Wolf, Julien Mille, and Greg Mori. Object level visual reasoning in videos. In *ECCV*, 2018.
- [4] João Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.
- [5] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. In *ECCV*, 2018.
- [6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.
- [7] Deepti Ghadiyaram, Du Tran, and Dhruv Mahajan. Large-scale weakly-supervised pre-training for video action recognition. In *CVPR*, 2019.
- [8] Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan Russell. Actionvlad: Learning spatio-temporal aggregation for action classification. In *CVPR*, 2017.
- [9] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *CVPR*, 2018.
- [10] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask r-cnn. In *ICCV*, 2017.
- [11] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2016.
- [12] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.

- [13] Antoine Miech, Ivan Laptev, and Josef Sivic. Learnable pooling with context gating for video classification. *arXiv preprint arXiv:1706.06905*, 2017.
- [14] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *T-PAMI*, 2015.
- [15] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Neurips*, 2014.
- [16] Swathikiran Sudhakaran, Sergio Escalera, and Oswald Lanz. Lsta: Long short-term attention for egocentric action recognition. In *CVPR*, 2019.
- [17] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015.
- [18] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016.
- [19] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *CVPR*, 2019.
- [20] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017.
- [21] Zhongwen Xu, Yi Yang, and Alex G Hauptmann. A discriminative cnn video representation for event detection. In *CVPR*, 2015.
- [22] Linchao Zhu, Laura Sevilla-Lara, Du Tran, Matt Feiszli, Yi Yang, and Heng Wang. Faster recurrent networks for video classification. *arXiv preprint arXiv:1906.04226*, 2019.
- [23] Linchao Zhu, Zhongwen Xu, and Yi Yang. Bidirectional multirate reconstruction for temporal modeling in videos. In *CVPR*, 2017.
- [24] Linchao Zhu, Zhongwen Xu, Yi Yang, and Alexander G Hauptmann. Uncovering the temporal context for video question answering. *IJCV*, 2017.