MIL-UTokyo at THUMOS Challenge 2015

Katsunori Ohnishi, Tatsuya Harada
University of Tokyo
7-3-1 Hongo Bunkyo-ku, Tokyo Japan
{ohnishi, harada}@mi.t.u-tokyo.ac.jp

Abstract

This paper describes our method for the action classification task of the THUMOS Challenge 2015. As appearance features, we extract the frame features from the middle layer of pretrained Convolutional Neural Network and encode them with VLAD. As motion features, we extract HOG, HOF and MBH along improved dense trajectories and encode them with Fisher Vector. Appearance features are classified with averaging Passive Aggressive and motion features with SVM. We achieve 66.8% mean average precision on the validation set.

1. Introduction

This paper describes our recognition system and our result in the THUMOS Challenge 2015. The aim of this challenge is to recognize a large number of action categories from open source videos in a realistic setting. Detailed information about this challenge is on the challenge website [1].

2. System Pipeline

Figure 1 shows the pipeline of our system for the action recognition. In this section, we describe our appearance features and motion features as well as their classification methods.

2.1. Appearance Features

We investigate novel methods for appearance features. We apply each frame to the VGG net [4] pre-trained on the ImageNet 2012 dataset and use the state of the 6th layer as a frame feature. After that, we encode obtained features with VLAD [2]. There are several reasons that we employ VLAD to encode the frame features. First, VLAD can express not only the meaning of the feature space, but also the spatial extent of the feature space. Second, VLAD is stable for high-dimensional features since it calculates the representation by simply subtracting the codeword vectors from the features.

We use one-versus-rest averaging Passive Aggressive (aPA) [6] for motion features as classifier because SVM shows lower performance than aPA for appearance features.

It is time-consuming to extract features from all frames. So we reduce the frame rates of validation and test videos from 30 fps to 5fps. Since train videos are very short, we do not reduce frame rates.

2.2. Motion Features

As motion features, we extract HOG, HOF and MBH along improved dense trajectories (iDT) [7]. First, we apply PCA to these features and reduce each feature to 64 dimensions. After that, we encode obtained features with Fisher Vector (FV) [3] and apply power and L2 normalization to the encoded vectors.

We use one-versus-rest SVM for motion features as classifier because aPA shows lower performance than SVM.

2.3. Late Fusion

In order to combine several features, we simply sum up obtained scores from classifiers.

3. Experiment Results

In this section we describe our experimental results on validation dataset. Note that we do not use background data at all. We only use 13320 videos from UCF101 [5] for training on validation set.

3.1. Appearance Features

Averaging means that obtained features from each frame are averaged over frames. Table 1 shows that the mean average precision (mAP) of VLAD and aPA is 50.9%; an improvement of 5.2% over Averaging and SVM. Thus, we encode frame features with VLAD and classify with aPA.

Table 1. mAP of appearance features

	Averaging	VLAD
SVM	45.7	46.2
aPA	42.7	**50.9**

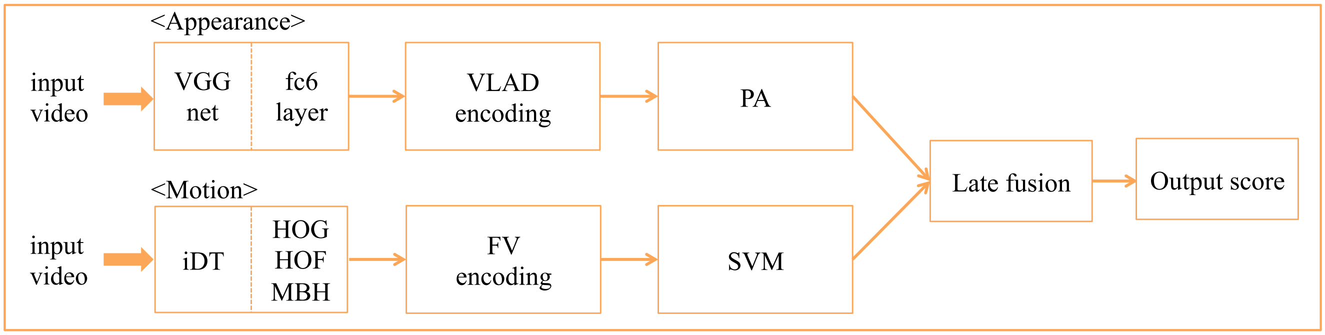

Figure 1. Our system pipeline

3.2. Motion Features

Table 2 shows mAP of HOG, HOF, MBH and all combined. When combining features, we simply sum up three scores obtained from classifiers. Unlike appearance features, aPA shows lower performance than SVM. Then, we encode each of HOG, HOF and MBH with FV, and classify each with SVM.

Table 2. mAP of motion features

	SVM	aPA
HOG	45.2	42.5
HOF	48.6	46.3
MBH	51.4	48.4
Combined (HOG+HOF+MBH)	**57.8**	55.6

3.3. Combination of Appearance and Motion features

In order to combine the results, we simply sum up the obtained scores. In fact, Table 3 shows that combining appearance representation with aPA and motion representation with SVM (proposed method) outperforms other methods.

Table 3. mAP of combining appearance and motion

Appearance	Motion	mAP
aPA	SVM	**66.8**
SVM	SVM	64.5
aPA	aPA	63.8

We also try slight setting change, that is, adding appearance features (Averaging and Maxpooling) to the proposed method. However, it does not outperform the method without adding as shown in Table 4.

Table 4. mAP of adding appearance features

	Avgeraging	Maxpooling	mAP
proposed method	-	-	**66.8**
proposed method+	SVM	-	64.5
	aPA	-	64.5
	-	SVM	65.8
	-	aPA	61.6
	SVM	SVM	63.2

Finally, the following is our proposed method in detail: first, we encode VGG features with VLAD and classify them with aPA, and encode HOG, HOF and MBH with FV and classify them with SVM. After obtaining scores from classifier, we simply sum up them. The confusion matrix of this method is shown in Figure 2.

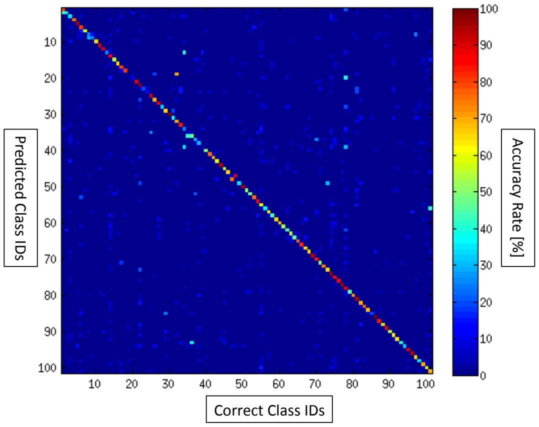

Figure 2. Confusion matrix on validation set

Acknowlagement

We would like to express our gratitude to Yusuke Mukuta of MIL-UTokyo for his help of extracting iDT.

References

[1] A. Gorban, H. Idrees, Y.-G. Jiang, A. Roshan Zamir, I. Laptev, M. Shah, and R. Sukthankar. THUMOS chal-

lenge: Action recognition with a large number of classes. http://www.thumos.info/, 2015. 1

[2] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, 2010. 1

[3] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007. 1

[4] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 1

[5] K. Soomro, A. Roshan Zamir, and M. Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. In *CRCV-TR-12-01*, 2012. 1

[6] Y. Ushiku, M. Hidaka, and T. Harada. Three guidelines of online learning for large-scale visual recognition. In *CVPR*, 2014. 1

[7] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, 2013. 1