TokyoTech-AIST at TRECVID 2017: Multimedia Event Detection Using Deep CNNs and Zero-Shot Classifiers

Nakamasa Inoue[1], Ryosuke Yamamoto[1], Na Rong[1], Satoshi Kanai[1], Junsuke Masada[1], Chihiro Shiraishi[1], Shi-wook Lee[2], and Koichi Shinoda[1]

[1]Tokyo Institute of Technology
[2]National Institute of Advanced Industrial Science and Technology, Japan
{inoue, ryamamot, na}@ks.cs.titech.ac.jp, {kanai, masada, siraisi}@ks.c.titech.ac.jp,
s.lee@aist.go.jp, shinoda@c.titech.ac.jp

1 Introduction

For multimedia event detection [1], we propose a system which employs support vector machine classifiers (SVMs) with features extracted by deep convolutional neural networks (CNNs) and zero-shot classifiers. This year, we introduce two types of classifiers: 1) CNN-SVM classifiers and 2) zero-shot classifiers. For CNN-SVM, GoogLeNet [2] is used to extract visual features from video data. SVMs for each event are trained with the visual features of 10 example videos in the official training dataset. For zero-shot classifiers, multiple CNN concept classifiers are linearly combined. Here, the similarity between an event name and each concept name measured by using word vectors is used to determine weights for the combination. In our experiments, three datasets are used to train CNNs: ImageNet [3, 4] for objects, Places [5] for scenes, and YFCC [6] for actions. Our best result was obtained by score fusion of the two types of classifiers. Our system achieved 52.9% for the AdHoc Events, and 15.3% for the Pre-specified Events in terms of Inferred Mean Average Precision.

2 Method

2.1 CNN-SVM Classifiers

Convolutional neural networks with support vector machines (CNN-SVM) are introduced to train classifiers for each event. Figure 1 (a) shows the overview of CNN-SVM. A video clip is first represented by a visual feature obtained by average pooling of deep features extracted from video frames. A detection score is then obtained from an SVM using the visual feature as its input. Note that a CNN for deep feature extraction is assumed to be trained on a large-scale dataset such as the ImageNet dataset [3], and SVMs are trained for each event by using the 10 example videos provided in MED training data.

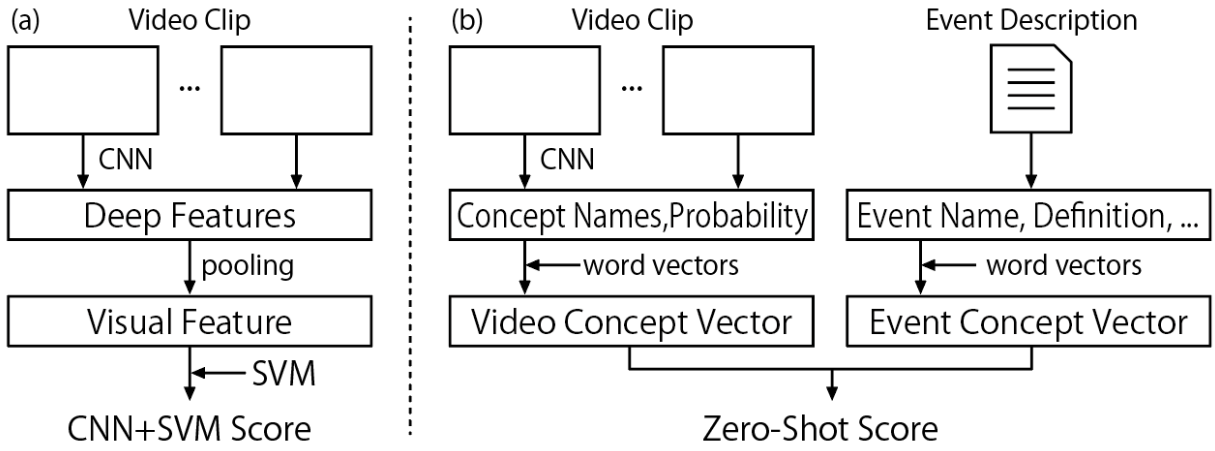

Figure 1: Overview of our system. (a) CNN-SVM Classifiers. (b) Zero-Shot Classifiers.

2.2 Zero-Shot Classifiers

Zero-shot classifiers are introduced to improve the robustness against data insufficiency in multimedia event detection. Figure 1 (b) shows the overview of our zero-shot approach. A detection score, namely a zero-shot score, is obtained by the cosine similarity between a video concept vector and an event concept vector, where a video concept vector is a vector representation of a video clip, and an event concept vector is a vector representation of the event description given in the MED dataset.

A video concept vector $v(V)$ is extracted from each video clip V with the following three steps. First, assuming that a pre-trained CNN is given on a concept dataset such as the ImageNet dataset [3], posterior probabilities $p_{i,c}$ for each concept c are calculated from video V, at every 2 seconds. Here, i is a frame index and n_V is a extracted frame number. Second, word vectors, which represent a word by a real-valued vector, are extracted from each concept name. The skip-gram model [7] is used to extract word vectors. Finally, the video concept vector is obtained by

$$v(V) = \sum_c \left(\frac{1}{|n_V|} \sum_{i=1}^{n_V} p_{i,c} \right) \phi(c), \tag{1}$$

where $\phi(c)$ is a word vector for a concept name c, i is a frame index, and $p_{i,c}$ is a posterior probability obtained from the CNN.

An event concept vector $u(E)$ for an event E is defined by a weighted sum of word vectors given by

$$u(E) = \sum_d \alpha_d \left(\frac{1}{|W_d(E)|} \sum_{w \in W_d(E)} \phi(w) \right), \tag{2}$$

where $\phi(w)$ is a word vector of a word w, d is a description type, $W_d(E)$ is a set of words in the description d of the event E, and α_d is a weight parameter. Note that since an event description for MED consists of seven description types of "Event Name", "Definition", "Explication", "Scene",

"Object/People", "Activities", and "Audio", the summation over d is for over them. To optimize the weight α_d, we evaluate each description type d independently on the MED-14 Kinderd Test dataset, and set $\alpha_d = \text{mAP}_d$. Here, mAP_d is Mean Average Precision on the evaluation for d, in which the event concept vector is made from only the words in $W_d(E)$.

Finally, a zero-shot score $s_{V,E}$ is obtained by cosine similarity between a video concept vector and and an event concept vector, i.e.,

$$s_{V,E} = \frac{v(V)^T u(E)}{\|v(V)\|\|u(E)\|}. \tag{3}$$

3 Datasets and Networks

Three datasets are used for pre-training of CNNs: ImageNet for objects, Places for scenes, and YFCC-Verb for actions.

ImageNet for Objects

The ImageNet dataset consists of 14 million images for 20K objects. The GoogLeNet [2], which is a convolutional neural network with 23 layers, is trained on this dataset. To estimate network parameters, ImageNet Shuffle [4], a method to train a network with unbalanced training samples, is introduced. We use a network with probabilistic outputs for 12,988 objects proposed in [4].

Places for Scenes

The Places dataset consists of 2.5 million images for scene recognition. The GoogLeNet is trained on this dataset. It has probabilistic outputs for 365 scenes such as "bridge", "cafeteria", and "campus".

YFCC-Verb for Actions

To train a network for actions, we make a training dataset, namely YFCC-Verb dataset, from the YLI-MED subset [8] of the YFCC100M dataset [6]. It consists of 50,000 videos with textual metadata of "Title", "User Tags", and "Descricription". However, since these metadata attached by users are often noisy, we introduce preprocessing to create verb labels for training.

There are four steps in the preprocessing. First, each text description is tokenized into words with their part of speech. Second, each word is lemmatized to get its base form. Third, verbs in the WordNet dictionary are selected, and the other words are eliminated. Finally, stop words and symbols are eliminated. As a result of the preprocessing, YFCC-Verb dataset has 4,126 verbs with 18,839 video clips.

A GoogLeNet is fine-tuned on the YFCC-Verb dataset. In fine-tuning, since one video clip can have more than one verb labels, a sigmoid layer with the cross entropy loss is used as the output layer.

Table 1: Mean Average Precision (%) for each run.

Run	MED-14 Kindred	MED-17 PS	MED-17 AdHoc
p-SVMbaseline	34.0	14.7	52.1
c-ActionVerb	28.4	9.1	-
c-Zeroshot	36.4	**15.3**	-
c-ZeroPlacemAPFusion	**38.1**	15.1	**52.9**

4 Experiments

4.1 Experimental Conditions

Deep features for CNN-SVM are extracted from the pool5/7x7_s1 layer of the GoogLeNet, at every 2 seconds of video data. The dimension of deep features is 1,024. For zero-shot classifiers, probabilistic outputs are obtained from the softmax layer, which has 12,988 and 365 dimensions for ImageNet-Shuffle [4], and Places 365 [5], respectively. Word2vec model contains 3 million words and transforms a word into a 300 dimensional vector. The model is pre-trained with GoogleNews [7]. For all experiments, these classifiers are trained on cluster servers with 4 NVIDIA Tesla P100 GPUs and 2 Intel Xeon E5-2680-V4 CPUs. Our runs in the official evaluation are summarized as follows.

p-SVMbaseline

This run uses only CNN-SVM classifiers. ImageNet is used to train a CNN.

c-ActionVerb

This run adds CNN-SVM classifiers on the YFCC-Verb dataset to the p-SVMbaseline.

c-Zeroshot

This run adds zero-shot classifiers for ImageNet objects to the p-SVMbaseline.

c-ZeroPlacemAPFusion

This run adds zero-shot classifiers for ImageNet objects and Places scenes to the p-SVMbaseline.

4.2 Results

Table 1 shows Mean Average Precision for each run. Our best results, 52.9% for MED-17 AdHoc Events and 15.3% for MED-17 Pre-specified (PS) Events, were ranked 2nd and 3rd respectively, among 4 and 6 participating teams. All results are shown in Figure 2.

We observe that zero-shot classifiers improve the performance; however, the classifiers were effective only for events related to many objects such as "Tuning a musical instrument" with "Guitar", "Piano", etc. With YFCC-Verb dataset, AP decreased for most of the events as shown in Figure 3. The GoogLeNet fine-tuned on YFCC-verb dataset did not improve the performance,

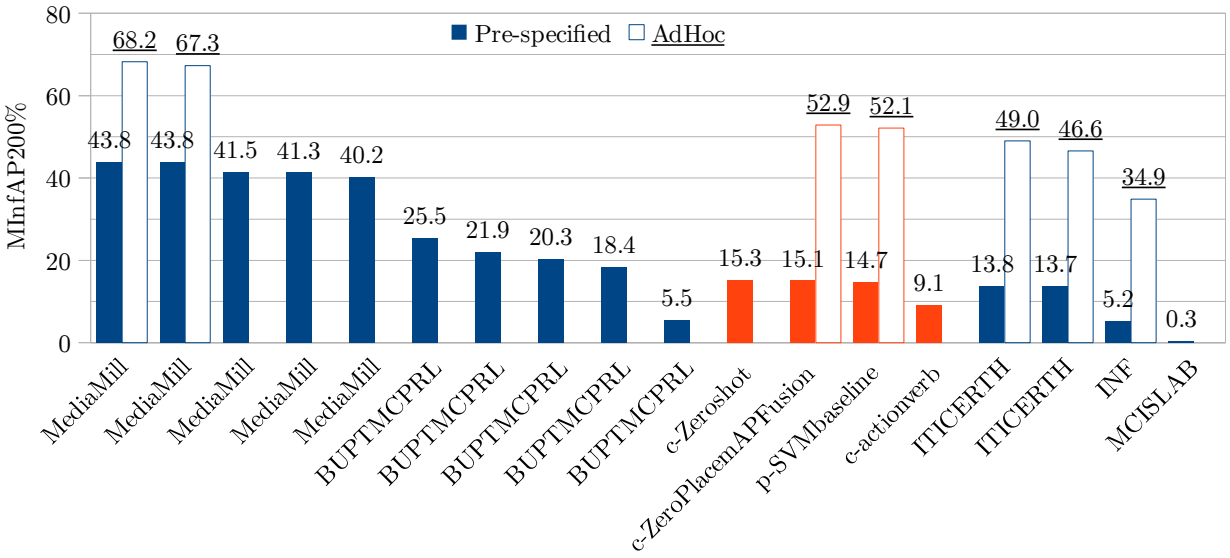

Figure 2: Results on MED17 Pre-specified and AdHoc Event Tests

because the number of training samples for each verb was imbalance. To train an effective network on this dataset, manual annotation is needed to attach more precise labels to video data.

5 Conclusion

We proposed a system using CNN-SVM classifiers and zero-shot classifiers for multimedia event detection. Our experiments showed that the two types of classifiers benefit from each other. Future work will focus on introducing recent zero-shot methods [9] and audio analysis for event detection.

Acknowledgement

This work was supported by JST CREST Grant Number JPMJCR1687, and JST ACT-I Grant Number JPMJPR16U5, Japan.

References

[1] G. Awad, A. Butt, J. Fiscus, D. Joy, A. Delgado, M. Michel, A. F. Smeaton, Y. Graham, W. Kraaij, G. Qunot, M. Eskevich, R. Ordelman, G. J. F. Jones, and B. Huet, "Trecvid 2017: Evaluating ad-hoc and instance video search, events detection, video captioning and hyperlinking," in *Proceedings of TRECVID 2017*, NIST, USA, 2017.

[2] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of CVPR*, pp. 1–9, IEEE, 2015.

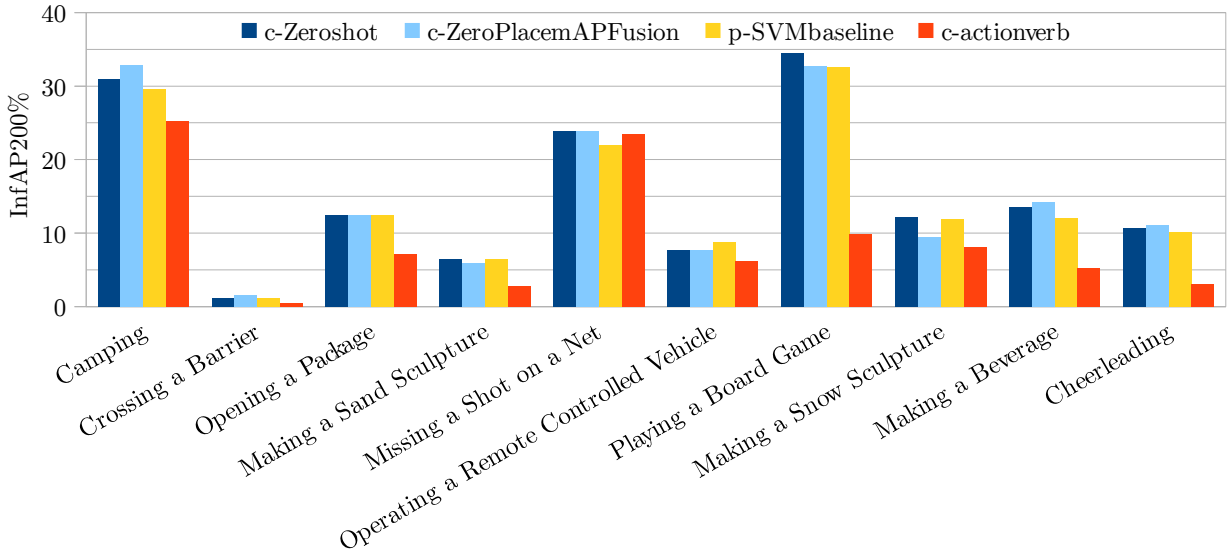

Figure 3: Average Precision by events.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of CVPR*, pp. 248–255, IEEE, 2009.

[4] P. Mettes, D. C. Koelma, and C. G. M. Snoek, "The imagenet shuffle: Reorganized pre-training for video event detection," in *Proceedings of ICMR*, pp. 175–182, ACM, 2016.

[5] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[6] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li, "Yfcc100m: The new data in multimedia research," *Communications of the ACM*, vol. 59, no. 2, pp. 64–73, 2016.

[7] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space." in *ICLR workshop*, 2013.

[8] J. Bernd, D. Borth, B. Elizalde, G. Friedland, H. Gallagher, L. Gottlieb, A. Janin, S. Karabashlieva, J. Takahashi, and J. Won, "The yli-med corpus: Characteristics, procedures, and plans," tech. rep., ICSI, 2015.

[9] N. Inoue and K. Shinoda, "Adaptation of word vectors using tree structure for visual semantics," in *Proceedings of ACM Multimedia*, pp. 277–281, 2016.