PKU_WICT at TRECVID 2021:

Instance Search Task

Yuxin Peng, Zhaoda Ye, Junchao Zhang, Hongbo Sun, Dejie Yang, and Zhenyu Cui

Wangxuan Institute of Computer Technology,

Peking University, Beijing 100871, China.

pengyuxin@pku.edu.cn

Abstract

In TRECVID 2021, we participated in all two types of Instance Search (INS) task, i.e., automatic way and interactive way. In automatic instance search, a two-stage approach of similarity calculation and re-ranking was applied. In our approach, action and person were recognized by deep neural networks separately, followed by score fusion for instance search. In *action-specific recognition*, it consisted of four modules, i.e., frame-level action recognition, video-level action recognition, object detection and facial expression recognition. In *person-specific recognition*, face detection, feature extraction and top *N* query expansion strategy were adopted. In *instance score fusion*, a re-ranking strategy was applied based on their recognition information, and then the recognition scores of person and action were merged for retrieval. In interactive instance search, the interactive query expansion strategy was conducted to refine the results from automatic search.

1. Overview

In TRECVID 2021[1], we participated in all two types of Instance Search (INS) task, namely automatic way and interactive way. 8 runs were submitted in total, i.e., 6 automatic runs and 2 interactive runs. The final official evaluation results in the Main task and Progress task are shown in Table 1. Table 2 presents a brief description of the symbols used in Table 1. The overall framework of our approach is presented in Figure 1.

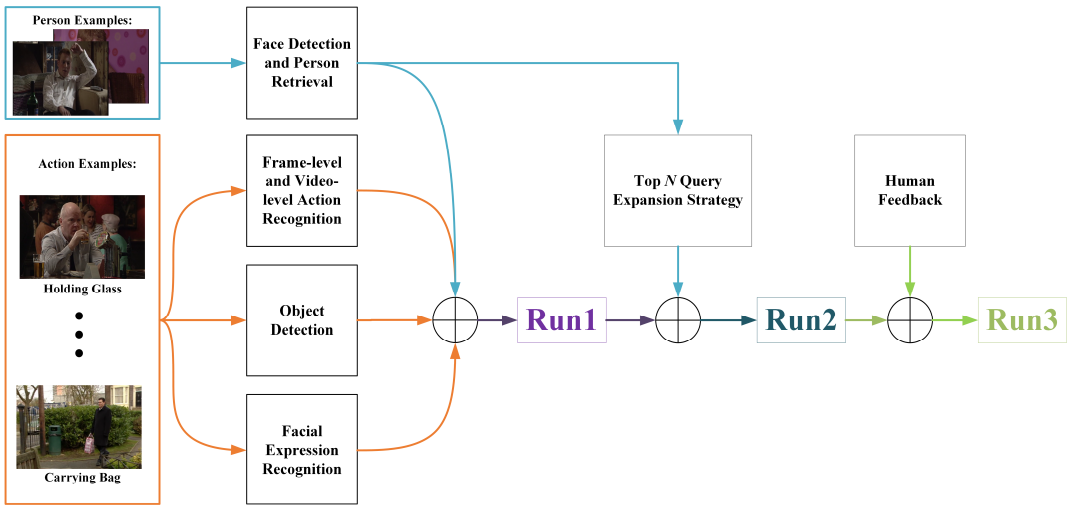

Figure 1: Overall framework of our approach.

In our submitted runs, the notation "A" indicates the provided video examples were not used, while "E" indicates the opposite. The notation "M" and "P" denote the Main task and the Progress task respectively. "Run2" comprises all the components utilized in the automatic instance search task, i.e., frame-level and video-level action recognition, object detection, facial expression recognition, deep face recognition and top *N* query expansion strategy. "Run1" excludes the top *N* query expansion strategy. And "Run3" denotes our interactive runs with human feedback.

Table 1: Results of our submitted 8 runs on Instance Search task of TRECVID 2021.

Task	Type	ID	MAP	Brief description
Main	Automatic	PKU_WICT_RUN2_ME	0.211	A+O+E+F+T
		PKU_WICT_RUN2_MA	0.200	A+O+E+F+T
		PKU_WICT_RUN1_ME	0.192	A+O+E+F
		PKU_WICT_RUN1_MA	0.183	A+O+E+F
	Interactive	PKU_WICT_RUN3_M	0.269	A+O+E+F+T+H
Progress	Automatic	PKU_WICT_RUN2_PE	0.209	A+O+E+F+T
		PKU_WICT_RUN2_PA	0.201	A+O+E+F+T
	Interactive	PKU_WICT_RUN3_P	0.270	A+O+E+F+T+H

Table 2: Description of our approach.

Abbreviation	Description
A	Frame-level and Video-level **A**ction Recognition
O	**O**bject detection
E	Facial **E**xpression Recognition
F	Deep **F**ace recognition
T	**Top** *N* Query Expansion Strategy
H	**H**uman Feedback

2. Our approach

2.1 Action-specific Recognition

There are 14 kinds of actions involved in both main and progress tasks, including "sit on couch", "holding cloth", "open door enter", etc. These actions are in various scenes and appearances, bringing great challenges for their recognition. The challenges are summarized as follows: (1) Some action categories are too similar to be distinguished. For example, the actions "holding glass" and "drinking" may both contain the content that people hold glass cups, which are easy to be confused. To distinguish them, the details whether people take the cups close to their mouth and drink the water should be captured. (2) Many non-action videos present similar content to action videos, which disturb the action recognition. For example, "holding paper" refers to the action that people

"hold" papers with their hands, while there are many videos where people place their hands on the paper, and these videos confuse the recognition model. (3) In addition, the provided training data is insufficient where there are only 4~6 video clips provided for each action category. However, the deep learning based methods need a large amount of training data.

Multiple methods for addressing the above challenges were designed and their ensemble results were adopted as the final action recognition. These methods included frame-level action recognition, video-level action recognition, object detection and facial expression recognition. For the training data problem, on the one hand, data from existing public datasets was collected for action recognition. On the other hand, images from the Internet were crawled to extend the training data.

Multiple methods were finally integrated to boost the action recognition performance, where prediction scores of a shot were fused as the final prediction score *ActScore*.

2.1.1 Frame-level Action Recognition

The frames extracted from the video describe the appearance of the action at a certain moment, which can be used to classify the actions without obvious motion, such as "sit on couch", "holding phone", etc. First, images from web and existing public datasets were collected to construct the training data. Then the image classification model SENet[2] was trained to predict the action score for each video frame. Finally, the frame-wise prediction scores in each video shot were merged to obtain the recognition result of the corresponding video.

(1) Training Data Collection

Training data was collected in two ways. First, images from the Internet were crawled. Since NIST officially provided detailed text definitions of the action categories, several keywords were selected for each category and searched the related web images via popular web search engines, including Baidu[1] and Bing[2]. Second, video frames of relevant datasets were exploited. Existing datasets, such as Kinetics-400[3], share several similar action categories with the INS task. So, frames extracted from Kinetics-400 videos were used to form the training data.

(2) Model Training and Predicting

For the challenge that some actions presented similar contents, fine-grained image classification technique was adopted for frame-level action recognition. Fine-grained image classification methods[4][5][6] can distinguish the fine-grained subcategories of the coarse-grained category, which can capture the discriminative details of images. Specifically, SENet was adopted as the frame-level action recognition model.

The recognition model was trained in a progressive way. A base SENet model was first trained with the collected web images and extracted video frames. Then the base model was utilized to predict the action categories for the video frames of INS database. According to the prediction scores,

[1] https://image.baidu.com
[2] https://cn.bing.com

frames with the top prediction scores were selected to extend the training data. Then the augmented training data was used to train the SENet model. The progressive training strategy gradually increased the number of training data, and made the training data distribution close to the INS database, which helped improving the recognition accuracy. The final well-trained SENet model was used to predict the classification score for each frame, and the maximal score of the frames in each shot was adopted to predict the shot category.

2.1.2 Video-level Action Recognition

Some actions are difficult to be distinguished with frame-level recognition because these actions depend on temporal and spatial changes, such as “open door enter” and “close door without leaving”. To effectively model the temporal and spatial information at the video level, the STNet[7], Non-local Network[8], 3D ResNets[9] and Video Swin Transformer[10] were adopted. For collecting sufficient training data, two video action recognition datasets of Kinetics-700[11] and Moments in Time[12] were merged.

(1) Training Data Collection

There are only 4-6 video clips for each category in the official data. It is hard to train and fit the model effectively only with the small amount of data. We investigated the existing public datasets for action recognition, checking the correlation between the action categories in the public datasets and those of the INS task. In the end, the Kinetics-700 and Moments in Time datasets were selected to expand the training data. Specifically, Kinetics-700 dataset contains 650,000 video clips and 700 human action categories, which can cover 11 relevant categories in the INS task. Moments in Time dataset includes 1 million labeled 3-second videos and 338 action categories, which can cover 12 relevant categories in the INS task. Tables 3 and 4 present the correspondences between action categories in INS task and the above two datasets, respectively. The data of the categories in the right columns was token as the training data.

(2) Model Training and Predicting

STNet, Non-local Network, 3D ResNets and Video Swin Transformer were adopted for the video-level action recognition. STNet exploits 2D and 3D convolutions to capture the local and global spatio-temporal information. Non-local network embeds non-local blocks in deep networks, which can capture long-range dependencies in the video. 3D ResNets is a deep convolutional neural network with spatial-temporal three-dimensional (3D) kernels with ResNet[13] as the backbone. Compared to 2D CNN, 3D ResNet adopts 3D convolutional layers to extract spatio-temporal information in videos. Video Swin Transformer uses the Vision Transformer[14] as the backbone and introduces the inductive bias of locality, which exploits the correlations among the pixels. Four models were trained on the merged dataset (Kinetics-700 and Moments in Time) and used them to classify all the video shots in INS database. Finally, the results obtained by the four models were combined to achieve better recognition results.

Table 3: Relations of categories between INS task and Kinetics-700.

Categories in INS task	Categories in Kinetics-700
holding phone	texting, look at phone
drinking	drinking shots, pouring beer, tasting beer, tasting wine, pouring wine,
laughing	laughing
holding paper	reading newspaper, shredding paper, making paper aeroplanes, folding napkins, folding paper, reading book, ripping paper
holding cloth	ironing, doing laundry, crocheting, hand washing clothes, sewing, folding clothes
kissing	kissing
open door enter	opening door, entering church
shouting	Throwing tantrum, arguing, crying
hugging	hugging not baby
closing door *wo* leaving	closing door
holding glass	opening wine bottle, pouring wine, breaking glass, opening bottle not wine

Table 4: Relations of categories between INS task and Moments in Time.

Categories in INS task	Categories in Moments in Time
sit on couch	sitting
holding phone	telephoning
drinking	drinking
laughing	laughing
holding paper	reading
kissing	kissing
open door enter/leave	opening
shouting	shouting
hugging	hugging
closing door *wo* leaving	closing
carrying bag	carrying

2.1.3 Object Detection

Although actions can be directly recognized by frame-level and video-level action recognition models, some unrelated videos are recalled at the same time. Some false positive videos do not contain the objects involved in the actions. Therefore, object detection methods were employed to

remove them. Table 5 presents the actions in INS task which involve objects. Related object categories were found in external object datasets, including MS-COCO[15], Visual Genome[16], and Object365[17]. MS-COCO includes 80 object categories, such as “bottle”, “couch”, etc. Visual Genome includes 33877 object categories, such as “door”, “book”, etc. Object365 includes 365 object categories, such as “sofa”, “paper towel”, etc. All of them are widely-used benchmarks for object detection.

Mask RCNN[18] and SENet-based Cascade RCNN[19] models were used to perform object detection, where the two Mask RCNN models were trained on MS-COCO and Visual Genome datasets, and SENet-based Cascade RCNN was trained on Object365 dataset. Object detection scores for each frame were obtained, then the maximal frame score was regarded as the detection score of the corresponding shot.

Table 5: Relations between actions and objects.

Actions in INS task	Objects in external datasets	External dataset
sit on couch	couch	MS-COCO
holding phone	cell phone	
drinking	bottle, wine glass, cup	
go up/down stairs	stairs	Visual Genome
holding paper	newspaper, book, papers, envelope	
open door enter	door, gate, door knob, door handle, garage door	
open door leave		
stand talk door		
close door *wo* leaving		
sit on couch	sofa	Object365
holding phone	cellphone, telephone, head phone	
drinking	bottle, cup, glasses, wine glass	
holding paper	book, towel/napkin, paper towel, tissue, notepaper	
carrying bag	handbag, backpack	

2.1.4 Facial Expression Recognition

Similar to the last year, facial expression recognition method was also used to help action recognition. Facial expression recognition is the task of identifying the expressions of human, such as anger and happiness. In INS task, the action “shouting” is hugely connected to human facial expression “angry”.

(1) Training Data Collecting

Data from both widely-used datasets and search engines was gathered to form the training data.

Concretely, data was collected from two datasets, CK+[20] and FER2013[21], which were dedicated to facial expression recognition. Meanwhile, to obtain more training data, an image crawler was employed to the image search engines, Baidu and Bing, with related keywords. Face detection model MTCNN[22] was utilized subsequently to crop human faces from the above images for constructing the training dataset.

(2) Model Training and Predicting

The VGGNet[23] model with 19 layers was chosen here as the facial expression recognition. The VGGNet model was fine-tuned with the aforementioned training data. During the predicting stage, human faces was first detected from video frames, and then took the cropped face images as input of the trained VGGNet model. Similarly, the maximal score of the frames in a shot was regarded as the final prediction score of the shot.

2.2 Person Identification

First, faces of person query examples were detected, filtered out abnormal "bad" faces and supplemented "good" faces with high detection confidence. Then, deep neural networks were utilized to extract features from query faces and shot faces, which were used to identify person. Besides, the top *N* query expansion strategy was adopted to enhance the query feature for specific person. Finally, the right person was retrieved based on the extracted face features and similarity calculation.

2.2.1 Face Detection

The pre-trained MTCNN[22] was used to detect faces of query person examples and shot key frames. Due to the camera angel or light intensity, sometimes faces are detected with bad quality, which were called "bad" faces. For getting better feature representation of specific person, "bad" faces were filtered out and "good" faces were replicated with high detection confidence as supplementation. By this means, the features of the query person faces were enhanced to facilitate the following person retrieval performance.

2.2.2 Person Retrieval

FaceNet model[24] was adopted to extract face features, which was pre-trained on the VGGFace2 dataset[25]. Based on the extracted face features, the similarity was calculated between query person examples and the gallery shot frames via cosine distance. The largest similarity between the shot i and person j was denoted $PerScore_{ij}$. In this way, the preliminary rank list for specific person was obtained.

2.2.3 Top *N* Query Expansion strategy

Relevant auxiliary information from the gallery shot frames was explored to boost the feature. The top *N* query expansion strategy was adopted. Based on the initial query features, the rank list of shots was obtained. Then, the mean feature of the top *N* nearest faces was adopted as the new query feature, which contained more discriminative information for specific person. The updated

query feature was closer to the real representation, which could improve the retrieval performance. After some rounds, the query feature became more robust. Then the similarity calculation was conducted to get the final person identification results.

2.3 Instance Score Fusion

In this section, the fusion strategies in re-ranking stage was introduced to fuse the prediction scores of action and person.

The first strategy is to search person based on candidate action shots. The candidate action shots were re-ranked according to the score s_1, which is defined as follows:

$$s_1=\mu \cdot PerScore \tag{1}$$

where μ is the reward parameter, and $PerScore$ is the prediction score mentioned in Section 2.2. The shots not in top M action-specific results were dropped by setting $\mu=0$, otherwise $\mu=1$.

The second strategy is to search specific action based on candidate person shots. Similar to the previous step, the candidate shots with persons were re-ranked according to the score s_2 as follows:

$$s_2=\mu \cdot ActScore \tag{2}$$

where μ is the parameter, and $ActScore$ is the prediction score mentioned in Section 2.1. The shots not in top N person-specific were dropped by setting $\mu=0$, otherwise $\mu=1$. Finally, the fusion score of a shot would be calculated as:

$$s_f=\omega(\alpha s_1+\beta s_2) \tag{3}$$

where α and β are weight parameters, and ω is a bonus parameter. We set $\omega>1$, if the shot simultaneously existed in the top N action-specific results and top-M person-specific results, otherwise $\omega=1$. Finally, fusion scores by preserving information of both action-specific recognition and person-specific recognition were obtained.

A time sequence based re-ranking strategy was proposed to refine the results, which can utilize the continuity of videos to adjust the scores of each shot based on adjacent frames information. Concretely, our approach recalculated the score of each shot by its neighbor shots' scores as follows:

$$s_f^{(i)}=\theta \sum\nolimits_{-T<k<T} s_f^{(i+k)}+(1-\theta)s_f^{(i)} \tag{4}$$

where $s_f^{(i)}$ denotes the score of i-th shot, T defines the range of the adjacent frames, and θ is a parameter to adjust the prediction score.

3. Interactive Search

The strategy for interactive search was based on RUN2_ME/ RUN2_PE and adopted the top N (N>1000) results as the candidate shots. First, the user returns the positive or negative labels for each topic's top-ranked results. Next, the positive samples (10 samples for each topic) were adopted to expand queries and compute the prediction scores of the other candidate shots. Finally, the scores based on expanded and original queries were fused to re-rank the candidate shots, where the negative samples were discarded.

4. Conclusion

Through the INS task in TRECVID 2021, we concluded that: (1) How to combine the recognition results of person and action is very important. (2) The approach has to distinguish the owner of the action in this year. (3) Human feedback can significantly boost the accuracy of INS task.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grants 61925201, 62132001 and 61771025. For the usage of BBC EastEnders video and image snapshots, we thank for the programme material copyrighted by BBC.

References

[1] George Awad, Asad A. Butt, Keith Curtis, Jonathan Fiscus, Afzal Godil, Yooyoung Lee, Andrew Delgado, Jesse Zhang, Eliot Godard, Baptiste Chocot, Lukas Diduch, Jeffrey Liu, Yvette Graham, Gareth J. F. Jones, and Georges Quénot. Evaluating Multiple Video Understanding and Retrieval Tasks at TRECVID 2021. Proceedings of TRECVID 2021, 2021.

[2] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation Networks. IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132-7141, 2018.

[3] João Carreira and Andrew Zisserman. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. IEEE Conference on Computer Vision and Pattern Recognition, pp. 6299-6308, 2017.

[4] Yuxin Peng, Xiangteng He, and Junjie Zhao. Object-Part Attention Model for Fine-grained Image Classification. IEEE Transactions on Image Processing, 27(3): 1487-1500, 2018.

[5] Tianjun Xiao, Yichong Xu, Kuiyuan Yang, Jiaxing Zhang, Yuxin Peng, and Zheng Zhang. The Application of Two-level Attention Models in Deep Convolutional Neural Network for Fine-grained Image Classification. IEEE Conference on Computer Vision and Pattern Recognition, pp. 842-850, 2015.

[6] Xiangteng He, Yuxin Peng, and Junjie Zhao. Which and How Many Regions to Gaze: Focus Discriminative Regions for Fine-grained Visual Categorization. International Journal of Computer Vision, 127(9): 1235-1255, 2019.

[7] Dongliang He, Zhichao Zhou, Chuang Gan, Fu Li, Xiao Liu, Yandong Li, Limin Wang, and Shilei Wen. StNet: Local and Global Spatial-temporal Modeling for Action Recognition. AAAI Conference on Artificial Intelligence, pp. 8401-8408, 2019.

[8] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local Neural Networks. IEEE Conference on Computer Vision and Pattern Recognition, pp. 7794-7803, 2018.

[9] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can Spatiotemporal 3D CNNs Retrace the History of 2d CNNs and ImageNet? IEEE conference on Computer Vision and Pattern Recognition, pp. 6546-6555, 2018

[10] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video Swin Transformer. arXiv:2106.13230, 2021.

[11] Lucas Smaira, Joao Carreira, Eric Noland, Ellen Clancy, Amy Wu, and Andrew Zisserman. A Short Note on the Kinetics-700-2020 Human Action Dataset. arXiv:2010.10864, 2020.

[12] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, and Aude Oliva. Moments in Time Dataset: One Million Videos for Event Understanding. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(2): 502-508, 2019.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778, 2016.

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv: 2010.11929, 2020.

[15] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. European Conference on Computer Vision, pp. 740-755, 2014.

[16] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li Jia-Li, David Ayman Shamma, Michael Bernstein, and Li Fei-Fei. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. International Journal of Computer Vision, 123(1): 32-73, 2017.

[17] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A Large-Scale, High-Quality Dataset for Object Detection. IEEE International Conference on Computer Vision, pp. 8430-8439, 2019.

[18] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask R-CNN. IEEE International Conference on Computer Vision, pp. 2961-2969, 2017.

[19] Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: Delving into High Quality Object Detection. IEEE Conference on Computer Vision and Pattern Recognition, pp. 6154-6162, 2018.

[20] Patrick Lucey, Jeffrey F. Cohn, Takeo Kanade, Jason M. Saragih, Zara Ambadar, and Iain A. Matthews. The Extended Cohn-kanade Dataset (ck+): A Complete Dataset for Action Unit and Emotion-specified Expression. IEEE Conference on Computer Vision and Pattern Recognition-Workshops, pp. 94-101, 2010.

[21] Challenges in Representation Learning: Facial Expression Recognition Challenge: https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data.

[22] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. IEEE Signal Processing Letters, 23(10): 1499-1503, 2016.

[23] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. International Conference on Learning Representations, pp. 1-14, 2015.

[24] Schroff F, Kalenichenko D, and Philbin J. FaceNet: A Unified Embedding for Face Recognition and Clustering. IEEE Conference on Computer Vision and Pattern Recognition, pp. 815-823, 2015.

[25] Qiong Cao, Li Shen, Weidi Xie, Omkar M. Parkhi, and Andrew Zisserman. Vggface2: A Dataset for Recognising Faces across Pose and Age. IEEE International Conference on Automatic Face & Gesture Recognition, pp. 67-74, 2018.