PKU_WICT at TRECVID 2022:
Disaster Scene Description and Indexing Task

Yanzhe Chen, HsiaoYuan Hsu, James Ye, Zhiwen Yang, Zishuo Wang,

Xiangteng He, and Yuxin Peng*

Wangxuan Institute of Computer Technology,

Peking University, Beijing 100871, China.

Abstract

In TRECVID 2022, we participated in two types of the Disaster Scene Description and Indexing (DSDI) task, i.e., LADI-based task (L task) and LADI + others task (O task), which are supported by the LADI (Low Altitude Disaster Imagery) dataset. **In the L task**, we proposed a two-stream approach considering both image and video, which consists of data cleaning, disaster-related feature extraction, and prediction score fusion. In the data cleaning stage, confident learning was applied to revise or discard the incorrect labels with the preliminarily trained models. In the disaster-related feature extraction stage, two modules were applied to fully take the advantage of both the image and video data in the training dataset, i.e., a pre-trained Swin-ViT model for image features, a C3D module for video features, with the ASL loss function to cope with the characteristics of the long-tailed distribution of LADI. In the prediction score fusion stage, the final prediction score of a test video was obtained by merging the prediction scores of multiple image/video models and then was utilized for retrieval. **In the O task**, the model structure remains the same, and extra data were collected on 16 categories, including flood, rubble, etc. Additional data for each category was used to train a model separately, and only the output of these corresponding categories participated in model ensemble. Our proposed approach achieved mAP scores of **46.8 in the L task** and **50.1 in the O task**, and the official evaluations showed that our proposed approach **ranked 1st in both L and O tasks**.

1 Overview

In TRECVID 2022[1], we participated in the Disaster Scene Description and Indexing (DSDI) task and submitted 8 runs in total: 4 for the LADI-based (L) training type and 4 for the LADI + Others (O) training type. The official evaluation results are shown in Table 1, and the illustration of our approach is shown in Figure 1. Moreover, the explanation of the brief descriptions in Table 1 is

[*] Yuxin Peng is the corresponding author (Email: pengyuxin@pku.edu.cn).

given in Table 2, expounding the methods and the details of ensembled models. In both subtasks, i.e., L and O, our team ranks 1st.

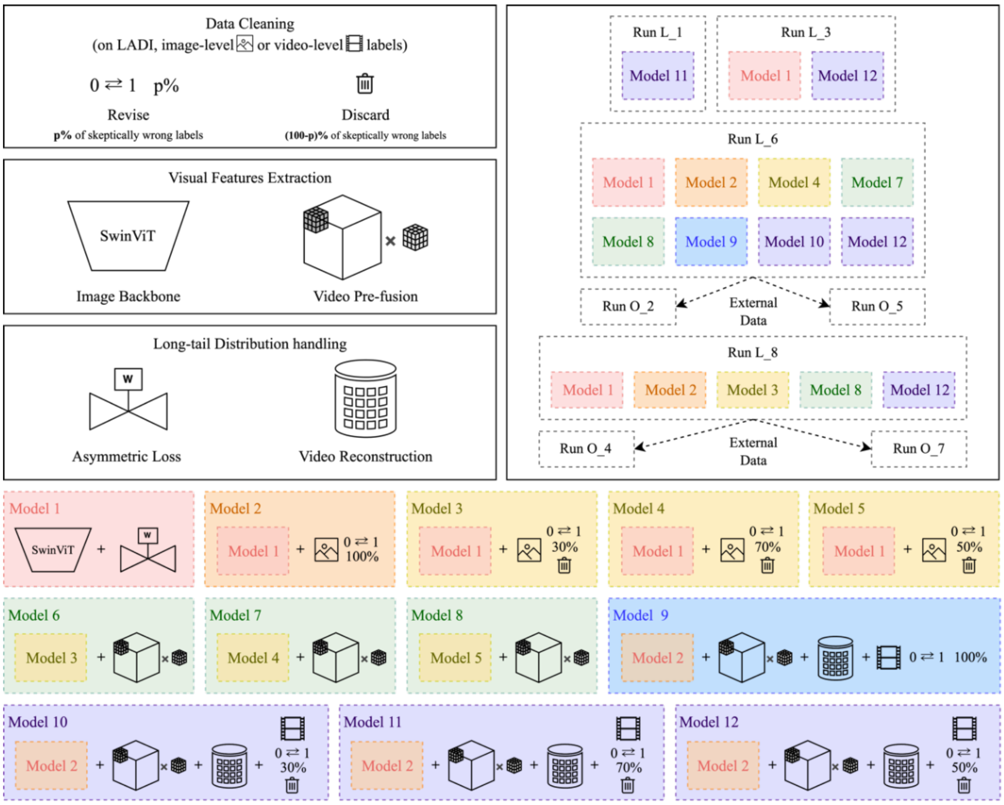

Figure 1: Framework of our approach for the 8 submitted runs.

Table 1: Results of our 8 submitted runs. M_i stands for different Models.

Type	ID	mAP	Brief description
L	L_PKU_WICT_1	0.4653	M_{11}
	L_PKU_WICT_3	0.4678	M_1+M_{12}
	L_PKU_WICT_6	**0.4680**	$M_1+M_2+M_4+M_{7\text{-}10}+M_{12}$
	L_PKU_WICT_8	0.4227	$M_{1\text{-}3}+M_8+M_{12}$
O	O_PKU_WICT_2	0.4995	Run6+S+T
	O_PKU_WICT_4	0.4819	Run8+S
	O_PKU_WICT_5	0.4287	Run6+S
	O_PKU_WICT_7	**0.5006**	Run8+S+T

Table 2: Description of our methods.

Abbreviation	Description
A	**A**symmetric loss
C	**C**3D pre-fusion model
R	**R**evising-based label cleaning
D	**D**iscarding-based label cleaning
V	Reconstructed **V**ideo labels
S	**S**ingle category O-finetuned models
T	**T**hirty-two categories O-finetuned model

In the L-type runs, only the official development dataset, Low Altitude Disaster Imagery (LADI), and past DSDI video testing sets were used for training. Since the annotation of the LADI dataset is crowd-sourced, several strategies[3] were adopted to eliminate noise in the dataset and correspondingly trained models for the ensemble. In detail, labels found to be problematic are either revised or directly discarded, noted as “R” or “D.” Besides the image-label pairs provided by the LADI dataset, images were further re-grouped into video frames regarding their metadata to construct video-label pairs, noted as “V”. Overall, first the cleaned image-label pairs were used to refine the image classifier, then the cleaned video-label pairs were used to supervise the additional model with video frames as input, which is implemented by a 3D CNN (i.e., C3D) and noted as “C.”

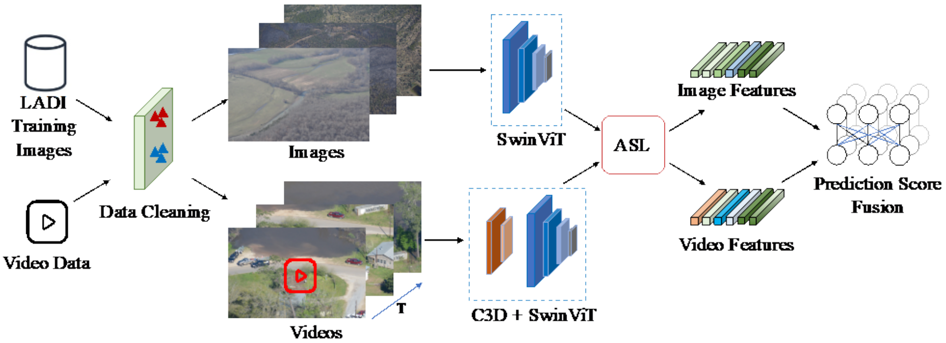

Figure 2: Pipeline of our approach.

The pipeline of our approach is shown in Figure 2, which consists of a C3D pre-fusion model and a SwinViT [2] as the video or image classifiers. We built a basic image classifier, model 1, without any tricks other than asymmetric loss noted as “A.” Models 2 to 5 were also image classifiers but with different image-level data cleaning procedures, including setting the ratios between “R” and “D” as 1:0, 3:7, 7:3, and 5:5. Based on model 3 to 5, we involved “C” and trained it on the past DSDI video testing sets to build model 6 to 8. Finally, “V” was involved in models 9 to 12, and thus all procedures mentioned above were gathered so far. Every L-type run took at least one of models 9 to 12, while in Run 3, 6, and 8, several pure image classifiers were also ensembled.

In the O-type runs, we resorted to 6 external remote sensing datasets as well as data crawled from the Internet to address the data imbalance issue between different categories in the LADI dataset. These additional data were used to finetune the models used in L-type runs. Specifically, we implemented 2 types of finetuning patterns, noted as “S” or “T.” Notation “S” means that only additional data of one single category was used for training, and the final model only predicted labels of this specific category. “T” means that additional data from all thirty-two categories were used for training, and the final model predicted multi-labels of all 32 categories together. By expanding two L-type runs, i.e., Runs 6 and 8, with “S” and “T”, we got four O-type runs.

2 Our Approach

2.1 Data Cleaning

Since the LADI dataset contains noise, confidence learning[3] (CL) was applied to clean the dataset. The model was first finetuned on the DSDI-2020 and DSDI-2021 testing sets and then performed the CL-based denoising on the LADI dataset. During the training stage, 3 different types of strategies were performed according to the confidence level: revise, discard, and hybrid. The meanings of each are described as follows:

1. Revise: The samples with low confidence in the training set are revised, i.e., converted between 0 and 1.
2. Discard: The samples with low confidence in the training set are discarded directly.
3. Hybrid: A portion of the samples with the lowest confidence are revised, while the rest of them are directly discarded.

2.2 Disaster-related Feature Extraction

2.2.1 Image Feature Extraction

The backbone model plays an important role in the performance of the DSDI task, and thus three different models were first tested to choose the most appropriate model, including EfficientNet-B5[4], ViT[5], and SwinViT[2]. The models were trained with the LADI dataset and evaluated on the DSDI-2021 testing set. The mAP results of each model are reported in Table 3. Since SwinViT is observed to own significant advantages compared to other models, it was adopted as our image backbone in the subsequent procedures.

Table 3: The mAP results of different backbones.

Backbone	**mAP**
EfficientNet-B5	23.62
ViT	25.49
SwinViT	27.97

2.2.2 Video Feature Extraction

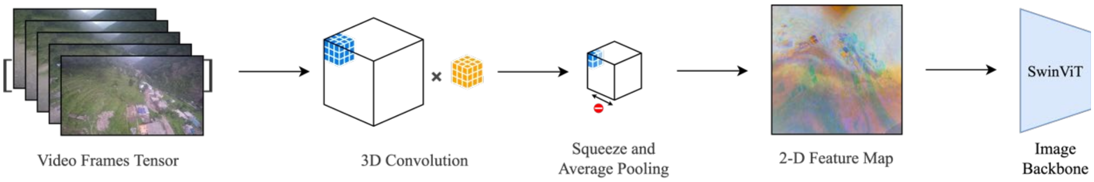

Figure 3: Pipeline of video-level feature extraction.

Our approach performed video-level feature extraction based on the image backbone, i.e., SwinViT, as shown in Figure 3. An intuitive idea for video classification is a bottom-up approach composed of two steps, classifying each extracted frame independently and integrating all frame-level results by pre-defined rules into one shot-level final result. However, since it takes the correlation between frames or temporal information under little consideration, it might miss some implicit clues to understand the video. To compensate for the deficiency while keeping its inherent superiority, a pre-fusion module was designed and attached ahead of the image classifier in the proposed approach, which fused the high-level visual features of frames with a neural network. Since frames are aligned by ascending time order before being concatenated for fusion, the temporal information can also be implicitly embedded. Specifically, the module was implemented by a 3D-CNN[6], i.e., C3D, that consists of two 3D convolutional layers of which output channel numbers are 64 and 1, respectively. The activation maps were squeezed along the channel axis and down-sampled by a 3D average pooling layer before inputting into the image classifier.

2.2.3 Loss Function

Since the LADI dataset shows a long-tail distribution, an appropriate loss function potentially alleviates the problem of unbalanced sample distribution in the dataset. ASL Loss[7] considers the sparsity of positive samples and pays more attention to positive samples in the training process, it was adopted in our proposed approach:

$$L_+ = (1-p)^{\gamma_+} \times \log(p) \tag{1}$$

$$L_- = p_m^{\gamma_-} \times \log(1-p_m) \tag{2}$$

where p is the network's output probability, and $p_m = \max(p-m, 0)$. $(1-p)^{\gamma_+}$ and $p_m^{\gamma_-}$ reduce the weight of the loss function on the high confidence samples, while hyperparameters γ_+ and γ_- act as smoothing factors. Therefore, the model pays more attention to the samples that are difficult to classify.

2.3 Prediction Score Fusion

After training models with different structures (i.e., image-level or video-level), confident learning settings (i.e., dropping-based or flipping-based), and hyper-parameters (i.e., learning rates and weight decay), their output probabilities were fused to achieve a more balanced effect across the 32 categories. Specifically, each model was assigned a weight among $\{0, 1, 2\}$. Before summing them up, the linear normalization was conducted to make sure the output of different models was in the same range of 0 to 1. The normalization is at two granularities: one is for the prediction of all 32 categories, and another is for every single category.

The weighted average of all models was calculated as the final output. In this way, different models complemented each other and improved the accuracy of the final prediction. Figure 4 shows the mAP results on each category of L-type runs, while categories 3, 4, 9 and 13 are omitted, for there are no samples in the official testing set this year.

3 LADI + Others (O)

The strategy for O task is based on the pipeline in the L task, which shares the same data cleaning and disaster-related feature extraction models. The L-type and O-type runs differ mainly in the dataset and prediction score fusion. Figure 5 shows mAP on each category of O-type run. For the data used in O task, images other than the provided LADI dataset were collected from two resources, i.e., public datasets and the Internet, which were annotated to form training data for our type O methods. Figure 6 shows the comparison of mAP results between the best O-type run, i.e., Run 7, and its corresponding L-type run, i.e., Run 8. Robust improvements can be observed in most categories, verifying the effectiveness of using the additional data adopted and processed.

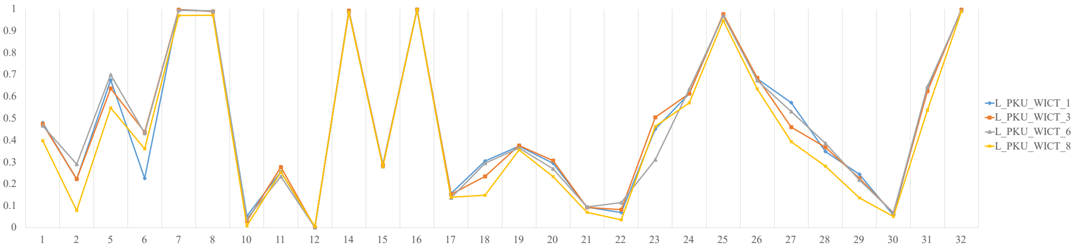

Figure 4: The mAP results on each category of L-type runs.

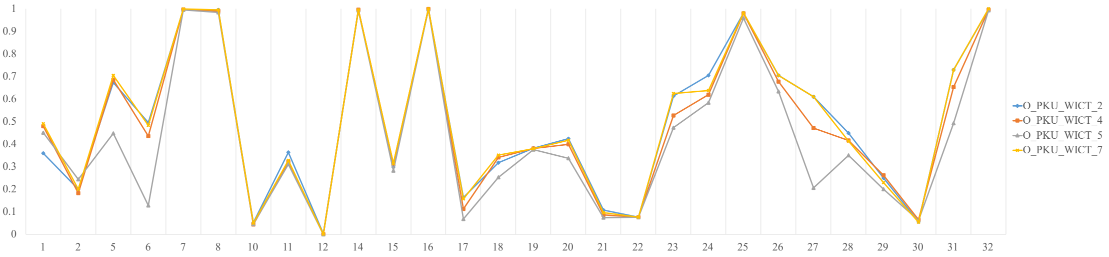

Figure 5: The mAP results on each category of O-type runs.

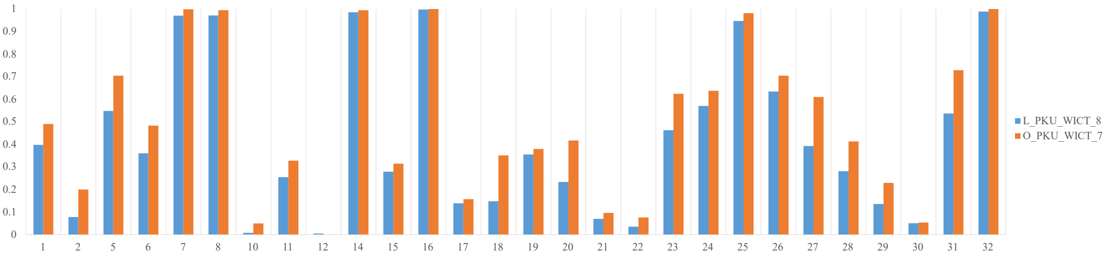

Figure 6: Comparison of mAP between the best L-type and O-type runs.

(1) Other Public Datasets

There are 32 different categories in the LADI dataset this year, while the numbers of images in different categories are severely imbalanced. This uneven data between categories is harmful to the accuracy of the deep models. Thus 6 external remote sensing datasets, UC Merced Land Use[8], NWPU-RESISC45[9], RSI-CB[10], AID[11], Massachusetts Building[12], and WHU-RS19[13][14], were resorted to acquire supplementary training data.

The image categories in the above remote sensing datasets are not exact matched in the LADI dataset. To address this issue, the data that matches each category separately were selected to compensate for the data deficiency in the LADI dataset. The additional data of the Top 5 categories with the least image numbers in LADI are presented in Table 4.

(2) Web Image Crawling

In addition to public remote sensing datasets, there are large quantities of images on the Internet that are easy to acquire. According to the official definitions of different categories, several keywords were utilized for each category to collect web images by Bing[1], one of the widely-used

[1] https://www.bing.com

search engines. After removing noisy images, there are about 300 images collected for categories that are not covered in the previous remote sensing datasets, such as landslide, washout, and rubble.

Table 4: The data addition of the Top-5 categories with least image numbers in LADI.

Categories in LADI	Image Numbers in LADI	Image Addition Numbers
snow/ice	115	2533
aircraft	142	360
pipe	481	198
railway	584	1660
water tower	594	2007

For the O task, three settings were applied to utilize the LADI dataset with the extra data, including all LADI data, randomly selecting the same amount of LADI data as the extra data, and using confidence learning to filter LADI data. Then the selected LADI data and the extra data were used to finetune a model for the corresponding category.

We had three models for every 16 categories in the O task, and each model only predicted the corresponding category's probability. Also, another model with all collected extra data was applied, which predicted probabilities of all 32 categories. The O models described above were ensembled with the averaged L model which outputs probabilities for all 32 categories.

Table 5: The weights of models for 16 categories in the O track.

Category	L Model	O Model for 32 Categories	O Model for Single Category
Flood	1	0	1
Rubble	1	0	1
Rock	1	1	1
Sand	1	1	1
Shrubs	1	0	1
Snow ice	1	1	1
Bridge	1	0	1
Dam levee	1	0	1
Pipe	1	1	1
Utility line	1	1	1
Railway	1	1	1
Boat	1	0	1
Cat	1	1	1
Truck	1	0	1
Lake	1	1	1
Ocean	1	1	1

For each category, five weights were needed to be determined, including the weights of the L model, the O model for all categories, and the three O models for the single category. They were set to 1 or 0, indicating to be fused or dropped. After the weights were determined, each model was weighted over the output of the category. The details of the weight settings for the 16 categories are shown in Table 5.

4 Conclusion

In TRECVID 2022, we participated in the Disaster Scene Description and Indexing (DSDI) task and ranked 1st in two subtasks. This paper presents our proposed approach and the official evaluation of its effectiveness. For alleviating the noise in the development set, several data-cleaning strategies were adopted. As for extracting visual features, a powerful image backbone, SwinViT, with a pre-fusion module, C3D, was adopted. Moreover, in dealing with the long-tail distribution of LADI, the asymmetric loss function (ASL) and additional data (only in O-type runs) were adopted.

ACKNOWLEDGMENTS

This work was supported by the grants from the National Natural Science Foundation of China (61925201, 62132001, U21B2025) and the National Key R&D Program of China (2021YFF0901502).

References

[1] G. Awad, A. Butt, K. Curtis, et al., “An overview on the evaluated video retrieval tasks at TRECVID 2022”, Proceedings of TRECVID 2022. 2022.

[2] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, Baining Guo, “Swin transformer: Hierarchical vision transformer using shifted windows”, Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). 2021: 10012-10022.

[3] Curtis Northcutt, Lu Jiang, Isaac Chuang, “Confident learning: Estimating uncertainty in dataset labels”, Journal of Artificial Intelligence Research (JAIR), 2021, 70: 1373-1411.

[4] Mingxing Tan, Quoc Le, “Efficientnet: Rethinking model scaling for convolutional neural networks”, International conference on machine learning (ICML), 2019: 6105-6114.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale”, arXiv preprint arXiv:2010.11929, 2020.

[6] Shuiwang Ji, Wei Xu, Ming Yang, Kai Yu, “3D convolutional neural networks for human action recognition”, IEEE transactions on pattern analysis and machine intelligence (TPAMI), 2012, 35(1): 221-231.

[7] Tal Ridnik, Emanuel Ben-Baruch, Nadav Zamir, Asaf Noy, Itamar Friedman, Matan Protter, Lihi Zelnik-Manor, “Asymmetric loss for multi-label classification”, Proceedings of the

IEEE/CVF International Conference on Computer Vision (ICCV). 2021: 82-91.

[8] Yi Yang, Shawn Newsam, “Bag-of-visual-words and spatial extensions for land-use classification”, Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems (ACM SIGSPATIAL). 2010: 270-279.

[9] Gong Cheng, Junwei Han, Xiaoqiong Lu, “Remote sensing image scene classification: Benchmark and state of the art”, Proceedings of the IEEE, 2017, 105(10): 1865-1883.

[10] Haifeng Li, Xin Dou, Chao Tao, Zhixiang Hou, Jie Chen, Jian Peng, Min Deng, Ling Zhao, “RSI-CB: A large scale remote sensing image classification benchmark via crowdsource data”, arXiv preprint arXiv:1705.10450, 2017.

[11] Gui-Song Xia, Jingwen Hu, Fan Hu, Baoguang Shi, Xiang Bai, Yanfei Zhong, “AID: A benchmark data set for performance evaluation of aerial scene classification”, IEEE Transactions on Geoscience and Remote Sensing (TGRS), 2017, 55(7): 3965-3981.

[12] Mnih Volodymyr, “Machine learning for aerial image labeling”, University of Toronto, 2013.

[13] Gui-Song Xia, Wen Yang, Julie Delon, Yann Gousseau, Hong Sun, Henri Maître, “Structural high-resolution satellite image indexing”, ISPRS TC VII Symposium-100 Years ISPRS (ISPRS). 2010, 38: 298-303.

[14] DengxinDai, Wen Yang, “Satellite image classification via two-layer sparse coding with biased image representation”, IEEE Geoscience and remote sensing letters (GRSL), 2010, 8(1): 173-176.

[15] Jeffrey Liu, David Strohschein, Siddharth Samsi, Andrew Weinert, "Large Scale Organization and Inference of an Imagery Dataset for Public Safety", 2019 IEEE High Performance Extreme Computing Conference (HPEC). IEEE, 2019: 1-6.