Hitachi at TRECVID DSDI 2020

Soichiro Okazaki[1] Quan Kong[1] Martin Klinkigt[1,2 *] Tomoaki Yoshinaga[1]
[1]Lumada Data Science Lab., Hitachi, Ltd.
[2]Future Design Lab, Minatomirai Research Center, KYOCERA Corporation
{soichiro.okazaki.xs, quan.kong.xz, tomoaki.yoshinaga.xc}@hitachi.com
martin.klinkigt.gt@kyocera.jp

Abstract

In this paper, we describe our approach for TRECVID 2020 DSDI task. This task requires to propose a system to output a ranked list of the top-k video clips that include the given disaster features. We treat this task as a multi-label multi-class classification problem by assigning predicted disaster features for the given frames and aggregating the frame-level labels to the video clip level labels for listing the ranked clip level result. LADI is used as our training dataset, that is a large scale disaster aerial image dataset with long-tail distribution, high-resolution and multiple noisy labels per image. To utilize LADI with consideration of handling the above characteristics, we propose a solution consisted of three parts: (1) Label encoding for smoothing the multiple annotations to reduce the noisy label propagation. (2) Incorporating a cost function based on Focal Loss for tackling the imbalanced data distribution. (3) Leveraging recently proposed efficient network architectures for dealing with high-resolution images as input. Furthermore, we combine these techniques with team NII-ICT AutoML solution, and also report the fusion results of ours with team NIIICT and NII_UIT, that reached a top mAP with 0.383 under the evaluation setting as training data with LADI-only track.

1. Introduction

In recent years, many large-scale visual recognition competitions (e.g. PASCAL VOC challenge [4], ILSVRC [19], Microsoft COCO Challenge [12], Google Landmark Challenge [26]) have been held vigorously all over the place. Among these image competitions, TRECVID (TREC Video Retrieval Evaluation) [1] has a long history as a competition that started in 2001 and has continued to this day. This large-scale video retrieval evaluation competition is hosted by NIST (National Institute of Standards and Technology) and the purpose of this competition is to encourage research in information retrieval by providing a large train/test dataset, uniform scoring procedures, and a forum for organizations interested in comparing their results.

Figure 1. A sample image from the LADI dataset. This image is labeled by 6 annotators as [flood/water, flood/water, smoke/fire, flood/water, flood/water, damage: none]. We pre-processed these labels into one soft-label with normalization. In the soft-label vector, each class has a following value as ground truth confidence: {flood/water class,1.0}, {smoke/fire,0.25}, {damage:none,0.25}, the rest 34 classes are 0.

From this year, TRECVID launched a new task entitled DSDI (Disaster Scene Description Indexing), which required to propose a system to output a ranked list of the top-k video clips that include the given disaster feature. We developed a system with a multi-label multi-class classification model, that assign predicted disaster features for the given frames and aggregate the frame-level labels to the video clip level labels for listing the ranked clip level result.

In this task, a dataset LADI (Low Altitude Disaster Imagery) [13] that contains 638K aerial images are provided for using. In LADI, there are 40K images labeled by AMT (Amazon Mechanical Turk) workers to support the development of computer vision capabilities for aerial images anal-

*Work done at Hitachi, Ltd. Currently at KYOCERA Corporation, Japan

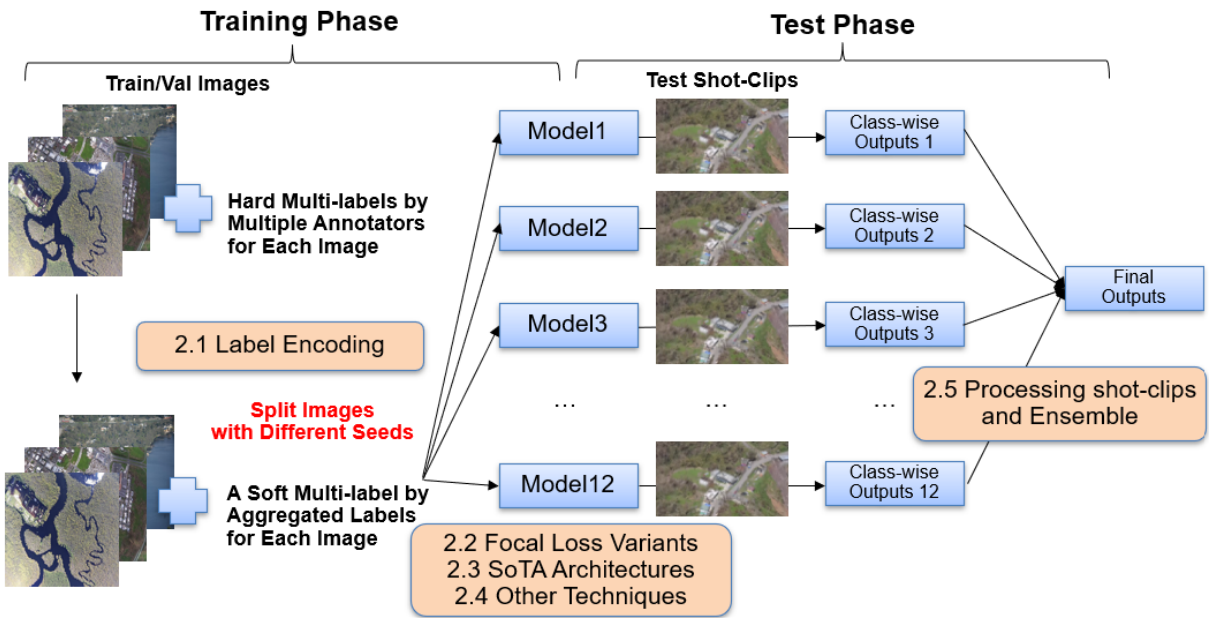

Figure 2. Overview of our pipeline for TRECVID DSDI 2020.

ysis. LADI is a large scale disaster aerial image dataset with *long-tail distribution for 37 classes*, *high-resolution* and *multiple noisy labels per image*.

To deal with these characteristics in our system, we developed a solution that incorporates three techniques for training the multi-label multi-class classifier: (1) Label encoding for smoothing to reduce the noisy label propagation. (2) Incorporating a cost function based on Focal Loss for tackling the imbalanced data distribution. (3) Leveraging recently proposed efficient network architectures for dealing with high-resolution images as input. In this paper, we describe these techniques and show the effectiveness of our solution in the experiment.

The features of our system can be concluded as two folds:

1) We developed a multi-label multi-class classification model to assign multiple disaster features to sampled frames from the video clip for predicting clip level feature index.

2) We proposed to train the classifier under the smoothed soft labels with a Focal Loss based cost function for dealing with class imbalanced problem in the train set, and further improve the performance by a score level model ensemble.

2. Proposed Solution

We first train a multi-label multi-class classifier for predicting the disaster feature index to the given frame. Then we sampled the frames from the given disaster video clip and predict the confidence score for each disaster feature index of the sampled frames by using the trained classifier. Video clip level confidence score for the given feature index is performed by fusing the frame level scores of it. Finally, top-k video clips that include the given disaster feature are sorted by using the clip level confidence scores.

Thus our system can be divided into two phases, a multi-label multi-class classifier training phase (Section 2.1-2.4), and a top-k video clip ranking phase (Section 2.5) for the given disaster feature index. The overview of our system is shown in Fig. 2.

2.1. Label Encoding for Noisy-annotated Labels

In LADI dataset, each image is annotated by multiple AMT workers and has multiple annotation labels. A sample image with annotated labels is shown in Fig. 1. In a multi-label dataset, however, a noisy-label problem often occurs because annotators cannot correctly label all classes when the class number is large. Such noisy-label problems are also found in LADI dataset. Therefore, for suppressing the annotation noise, we preprocessed annotated labels into one soft-label for each image by counting the number of occurrences for each class and using these numbers as ground truth confidence. From this method, we can simultaneously mitigate the mislabeled class (e.g. "damage: none" in Fig. 1) and capture the unconfident label class (e.g. "damage: smoke" in Fig. 1) as low confidence. For normalization, the values of the soft-label vector are created from being divided by the max number of occurrences in the vector. We use these smoothed soft labels vectors as ground truth confidence in our experiments.

2.2. Focal Loss Variants for Imbalanced Dataset

When training with a class imbalanced dataset, scarce classes cannot be captured well with normal cross-entropy

loss, because the learned models are highly biased towards abundant classes. Therefore, many approaches have been proposed to solve the problem by assigning weights to scarce classes.

Focal Loss [11] is one of the approaches proposed in object detection field. In Focal Loss, the losses of scarce classes are weighted with a hyper-parameter, and we can simultaneously learn abundant classes and scarce classes by controlling the hyper-parameter. In addition to the Focal Loss, we adopted two variants cost function of Focal Loss for our systems: (1) Class-Balanced Loss [3], which theoretically considers the weighting balance between abundant classes and scarce classes. (2) Reduced Focal Loss [20], which is designed to control the accuracy trade-off between scarce classes and abundant classes using different loss curves.

Moreover, the original Focal Loss is designed for learning with single ground truth label per sample, thus we have modified the Focal Loss function to be adapted for being capable of treating soft multi-label learning as follows:

$$\mathrm{FL}(\hat{p}_i) = -\sum_{i}^{C}(1-\hat{p}_i)^{\gamma}\log\hat{p}_i, \quad (1)$$

$$where, \hat{p}_i = 1-|y_i-p_i| \quad (2)$$

where, p_i represents model's predicted sigmoid value for i-th class and y_i represents the soft ground truth label of i-th class for the given image. C is the number of disaster features (class number). γ is a tunable *focusing* parameter used in Focal Loss. For Class-Balanced Focal Loss(CBFL) and Reduced Focal Loss(RFL), we performed the similar modification as follows:

$$\mathrm{CBFL}(\hat{p}_i) = -\sum_{i}^{C}\frac{1-\beta}{1-\beta^{n_y}}(1-\hat{p}_i)^{\gamma}\log\hat{p}_i \quad (3)$$

$$\mathrm{RFL}(\hat{p}_i) = -\sum_{i}^{C}(w_1\log\hat{p}_i + w_2(\frac{1-\hat{p}_i}{th})^{\gamma}\log\hat{p}_i \quad (4)$$

$$where, w_1 = \begin{cases} 1 & (p_i > th) \\ 0 & (p_i <= th) \end{cases} \quad (5)$$

$$w_2 = \begin{cases} 1 & (p_i <= th) \\ 0 & (p_i > th) \end{cases} \quad (6)$$

where, β, th, n_y are the hyper-parameters used for controlling the accuracy trade-off between the abundant classes and the scarce classes. The frequency n_y of each class used in Class-Balanced Loss are calculated based on the number of samples of all 37 classes from all 40K annotated images shown in Table 1. We trained various models with different hyper-parameters for controlling the accuracy trade-off between the abundant classes and the scarce classes.

Table 1. Class number for all 32 classes and 5 none-classes. These numbers are used in Class-Balanced Focal Loss as n_y.

class	number
damage (misc)	24009
flooding / water damage	33120
landslide	2237
road washout	3613
rubble / debris	18186
smoke / fire	1756
dirt	11866
grass	19623
lava	67
rocks	1067
sand	2565
shrubs	13513
snow/ice	116
trees	21183
bridge	3132
building	15133
dam / levee	616
pipes	554
utility or power lines / electric towers	10612
railway	742
wireless / radio communication towers	1136
water tower	749
road	16333
aircraft	201
boat	2500
car	13846
truck	7579
flooding	4077
lake / pond	5879
ocean	3029
puddle	2386
river / stream	7362
damage: none	98412
environment: none	6748
infrastructure: none	11814
vehicle: none	15420
water: none	13902

2.3. Backbones for High-Resolution Images

We use three network architectures as backbones: ResNeSt [29], HRNetV2 [21, 22, 25], and RegNet [17, 18]. In high-resolution images, an effective attention mechanism is crucial for creating a good recognition system. There are many proposed architectures which incorporating spatial attention mechanism (e.g. SENet [7], Attention Branch Network [5], ResNeSt [29]). In our experiments, we have tried many backbone architectures [5–7, 9, 14, 18, 22, 23, 27, 29] and ResNeSt backbone got the best accuracy in all architectures. Thus we selected ResNeSt as a default backbone.

Table 2. Summary of our created models. γ, β, th are the hyper-parameters of each imbalanced loss. LRAP is a label ranking AP score of local validation dataset, created by scikit-learn [16]. LRAP score is just a reference, as each model is trained with different training split. We use Model A01-C12 for creating our final ensemble models. Model-D13 is used as NII_UIT 2, expecting domain adaptation effect.

Index	Architecture	Focal	Class-Balanced	Reduced	Other Techniques	LRAP
A01	RegNet-6.4F	$\gamma : 2.0$	-	-	-	0.846
B02	**HRNet-W18**	$\gamma : 2.0$	-	-	**Input resolution: 448*448 (default: 360*360)**	**0.854**
C03	ResNeSt-50d	$\gamma : 2.0$	$\beta : 0.999$	-	-	0.844
C04	ResNeSt-50d	$\gamma : 2.0$	$\beta : 0.9999$	-	-	0.853
C05	ResNeSt-50d	$\gamma : 2.0$	-	th : 0.3	-	0.852
C06	ResNeSt-50d	$\gamma : 2.0$	-	th : 0.5	-	0.844
C07	ResNeSt-50d	$\gamma : 2.0$	-	th : 0.7	-	0.846
C08	ResNeSt-50d	$\gamma : 2.0$	-	-	Multiply the losses of none-classes by 0.1	0.837
C09	ResNeSt-50d	$\gamma : 2.0$	-	-	Multiply the losses of none-classes by 0.01	0.825
C10	ResNeSt-50d	$\gamma : 2.0$	-	-	Snapshot Ensemble of Model-C09	0.823
C11	ResNeSt-50d	$\gamma : 2.0$	-	-	Outlier removal technique	0.843
C12	ResNeSt-50d	$\gamma : 2.0$	-	-	Symmetric Lovasz Loss with square root scaling	0.849
D13	SE-IBN-ResNet50	$\gamma : 2.0$	-	-	-	0.824

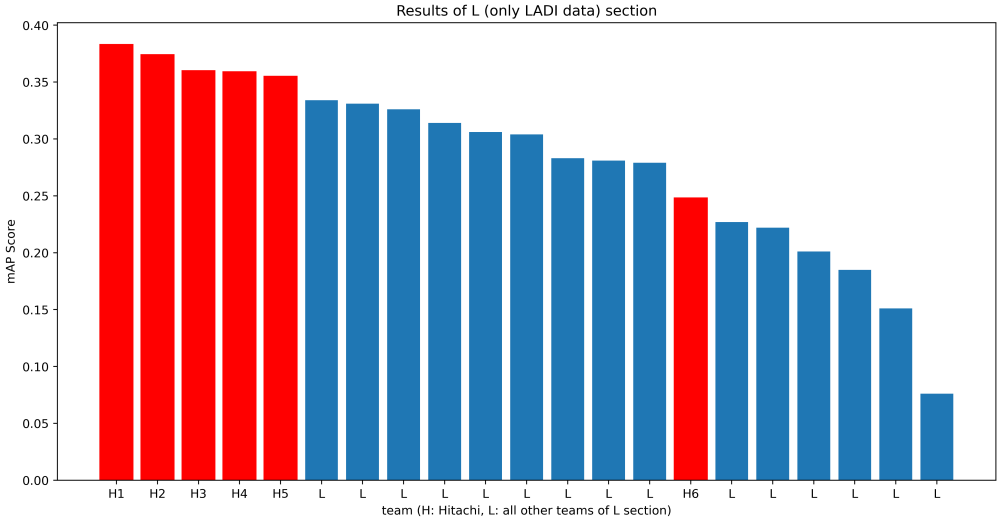

Figure 3. Results of L (only LADI data) section. Our submissions rank as 1st-5th and 15th places in all 21 submissions.

Furthermore, these aerial images sometimes include small object classes (e.g. car, boat, building, bridge). Therefore, it is necessary to recognize each tiny object from high-resolution images. For recognizing these high-resolution images, there are many proposed architectures (HRNetV1 [21, 25], HRNetV2 [22]) in object detection/segmentation tasks, which need more high-resolution accuracy than in classification tasks. From these models, we selected HRNetV2 as one backbone. In addition to ResNeSt and HRNetV2, we also selected RegNet backbone which controls depth/width/resolution scaling more effectively than EfficientNet [23]. In our experiments, We used these various backbones as listed in Table 2.

2.4. Other Training Techniques

In addition to the approaches mentioned in section 2.1-2.3, we have incorporated various training techniques for creating more diversity in ensembling. The approaches are as follows: (1) Snapshot Ensemble [8] (2) Symmetric Lovasz Softmax Loss [2,28] (3) Outlier removal technique (4) Multiplying the losses of 5 none-classes by a small value for recognizing 32 classes more efficiently. In this section, we especially describe our outlier removal technique.

In LADI dataset, there are some outlier images that have annotated labels like normal images. These sample images are shown in Fig. 5. Using these outlier images with annotated labels, however, have a bad effect on training, because

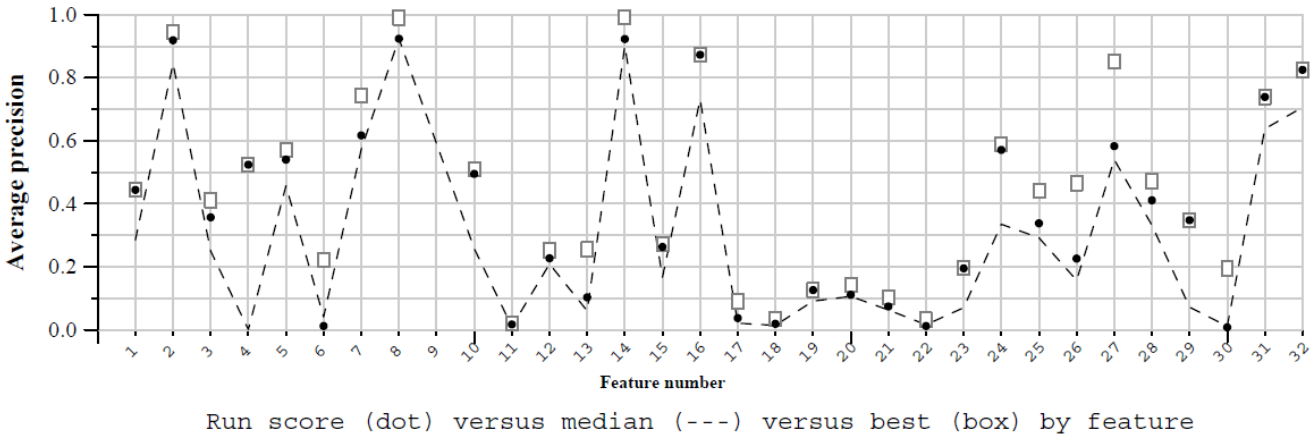

Figure 4. Detailed results of our 1st ranked submission (VAS 2). Class-wise mAP score are presented with all submissions' median/best mAP scores.

Figure 5. (above) sample normal images and (below) sample outlier images from LADI dataset. These images are selected from 40K labeled images.

these annotated labels will be incorrect since the outlier images represent no classes. Therefore, we introduced an automatic outlier removal technique. In automatic outlier removal, we always remove the max loss value in each mini-batch losses from the summation of losses. The idea behind this approach is as follows: If the models have learned each class correctly, the loss of each outlier image will have a high value. Using this approach, we created Model-C11 listed in Table 2.

2.5. Shot-clips Ranking and Ensemble Method

For processing test shot-clips, we encoded each shot-clip into image frames using ffmpeg with 10fps. After created frames from all test shot-clips, we produce the sigmoid outputs for all frames in a shot-clip using trained models, and selected max confidence value as the output for each feature in a shot-clip.

These shot-clip level confidence scores for the given feature index are merged by fusing the frame level scores of it. Finally, top-k video clips that include the given disaster feature are sorted by using the clip level confidence scores.

For ensemble, we created various models using techniques explained in section 2.1-2.4, and selected 10 or 12 models from the created models (See Table 3). After selected 10 or 12 models, the sigmoid outputs of these models are merged with equal weight for test shot-clip inference. In Table 2, we summarize the methods used in our experiments.

Table 3. Details of our submission results. All runs are performed under trained with only LADI dataset as TYPE *L*.

submission	mAP	description
VAS 2	0.383	10 models + NIIICT model (2:1)
NII_UIT 1	0.374	10 models + NIIICT model (5:1)
VAS 4	0.360	12 models ensemble
VAS 1	0.359	VAS 4 + VAS 3 ensemble
VAS 3	0.355	10 models ensemble
VAS LateSub1	0.338	Model-C03 (single model)
NII_UIT 2	0.248	Model-D13 (single model)

3. Experiments

In this section, we present our experimental setting and results.

Experimental setting: We conducted our experiments under the following setting: (1) Data augmentation: Resized to 360 $\times$ 360 (only for HRNet-W18, resized to 448 $\times$ 448), RandomHorizontalFlip, RandomVerticalFlip, RandomRotation, and ColorJitter with PyTorch [15] default parameters. (2) Train batch size: 28 (3) Epoch: 40 (for snapshot ensembling, we trained Model-C09 in Table 2 for 300

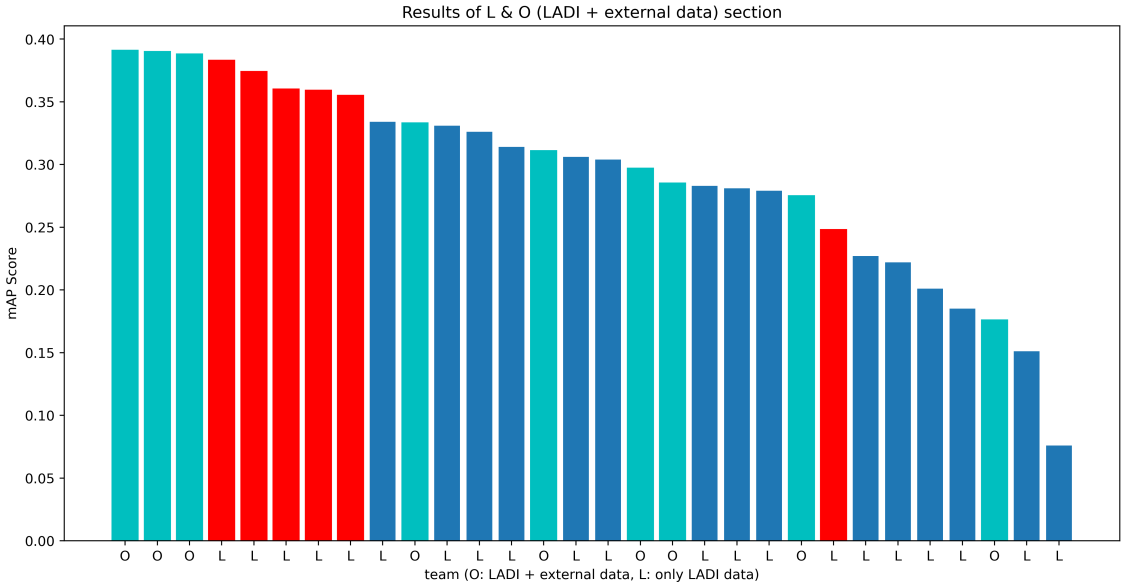

Figure 6. Results of L + O (LADI + external data) section. Our submissions rank as 4th-8th and 23th places in all 30 submissions.

epochs and selected the last epoch model as Model-C10.) (4) Dataset split ratio: 80% (train) and 20% (validation) (5) Other hyper-parameters: same as used in IBN-Net [14] GitHub repository[*1]. All our models are trained and tested on 1 GPU (GeForce RTX 2080Ti).

Results: Fig. 3 shows the L (LADI-only) section results and Table 3 shows the test scores for all of our submissions. In Table 3, "10 models" we used for ensemble are consist of Model A01-C04/C06-C09/C11-C12 described in Table 2. "12 models" are also an ensemble system by using the model from A01-C12 in Table 2. We collaborated with team NII_UIT and reported the fusion system's result as submission file NII_UIT 1, that also will be reported in team NII_UIT's paper [24]. Specifically, VAS 2 and NII_UIT 1 system are two fusion systems that consist of 10 models from VAS and a single model from NIIICT AutoML solution with different weighting parameter 2:1 and 5:1, respectively. The detail of NIIICT's solution can be referred in their reports [10]. Overall, 21 runs are submitted to L (only LADI data) section and our submissions rank as 1st-5th/15th places in all 21 submissions.

In Fig. 4, class-wise mAP scores of our 1st ranked submission (VAS 2) are presented with other teams' mAP scores. As a whole, our 1st ranked model achieved good performance in all 32 classes.

For reference, we present all teams' scores including O (LADI + external data) section in Fig. 6. In spite of using only LADI dataset, our models achieve competing results with the models trained with additional external datasets. In addition, we used only 40K labeled images and did not use other 598K unlabeled images in this competition.

*1: https://github.com/XingangPan/IBN-Net

4. Conclusion and Acknowledgement

In this paper, we presented our approach for TRECVID 2020 DSDI task. We used LADI as our training dataset, that is a challenge dataset with noisy-annotated labels, imbalanced samples per class and high-resolution images including tiny objects. Our proposed system is designed to overcome the above challenges, and the fusion system of ours and team NIIICT achieved a top accuracy under a training data with LADI-only track.

This work is a cooperation with team NIIICT and NII_UIT. We appreciate Shoichiro Iwasawa et al. from NIIICT for providing their AutoML solution as a part of the fusion system and also appreciate the advice and supports from Shin'ichi Satoh of team NII_UIT for the technical discussion and system submissions.

References

[1] George Awad, Asad Butt, Keith Curtis, Yooyoung Lee, Jonathan Fiscus, Afzal Godil, Andrew Delgado, Jesse Zhang, Eliot Godard, Lukas Diduch, Jeffrey Liu, Alan F. Smeaton, Yvette Graham, Gareth Jones, Wessel Kraaij, and Georges Quénot. Trecvid 2020: comprehensive campaign for evaluating video retrieval tasks across multiple application domains. In *Proceedings of TRECVID 2020*, 2020.

[2] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[3] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[4] Mark Everingham, Luc Van Gool, C. K. I. Williams, J. Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. In *International Journal of Computer Vision (IJCV)*, 2010.

[5] Hiroshi Fukui, Tsubasa Hirakawa, Takayoshi Yamashita, and Hironobu Fujiyoshi. Attention branch network: Learning of attention mechanism for visual explanation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[7] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[8] Gao Huang, Yixuan Li, Geoff Pleiss, Zhuang Liu, John E. Hopcroft, and Kilian Q. Weinberger. Snapshot ensembles: Train 1, get m for free. In *International Conference on Learning Representations (ICLR)*, 2017.

[9] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[10] Shoichiro Iwasawa, Kiyotaka Uchimoto, Yutaka Kidawara, and Shin'ichi Satoh. Is automl a practical way of tackling dsdi task? In *TRECVID notebook paper*, 2020.

[11] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[12] Tsung-Yi Lin, M. Maire, Serge J. Belongie, J. Hays, P. Perona, D. Ramanan, Piotr Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *IEEE European Conference on Computer Vision (ECCV)*, 2014.

[13] J. Liu, D. Strohschein, S. Samsi, and A. Weinert. Large scale organization and inference of an imagery dataset for public safety. In *IEEE High Performance Extreme Computing Conference (HPEC)*, 2019.

[14] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *IEEE European Conference on Computer Vision (ECCV)*, 2018.

[15] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Neural Information Processing Systems (NeurIPS)*, 2019.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. In *Journal of Machine Learning Research (JMLR)*, 2011.

[17] Ilija Radosavovic, Justin Johnson, Saining Xie, Wan-Yen Lo, and Piotr Dollár. On network design spaces for visual recognition. In *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[18] Ilija Radosavovic, Raj Prateek Kosaraju, Ross Girshick, Kaiming He, and Piotr Dollár. Designing network design spaces. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[19] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. In *International Journal of Computer Vision (IJCV)*, 2015.

[20] Nikolay Sergievskiy and Alexander Ponamarev. Reduced focal loss: 1st place solution to xview object detection in satellite imagery. In *arXiv preprint arXiv:1903.01347*, 2019.

[21] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for hu-

man pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[22] Ke Sun, Yang Zhao, Borui Jiang, Tianheng Cheng, Bin Xiao, Dong Liu, Yadong Mu, Xinggang Wang, Wenyu Liu, and Jingdong Wang. High-resolution representations for labeling pixels and regions. In *arXiv preprint arXiv:1904.04514*, 2019.

[23] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning (ICML)*, 2019.

[24] Hung-Quoc Vo, Tien-Van Do, Vinh-Tiep Nguyen, Thanh-Duc Ngo, Duy-Dinh Le, Zheng Wang, and Shin'ichi Satoh. Nii_uit at trecvid 2020. In *TRECVID notebook paper*, 2020.

[25] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2019.

[26] Tobias Weyand, Andre Araujo, Bingyi Cao, and Jack Sim. Google landmarks dataset v2 – a large-scale benchmark for instance-level recognition and retrieval. In *arXiv preprint arXiv:2004.01804*, 2020.

[27] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[28] Jiaqian Yu and Matthew B. Blaschko. Learning submodular losses with the lovasz hinge. In *International Conference on Machine Learning (ICML)*, 2015.

[29] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Zhi Zhang, Haibin Lin, Yue Sun, Tong He, Jonas Muller, R. Manmatha, Mu Li, and Alexander Smola. Resnest: Split-attention networks. In *arXiv preprint arXiv:2004.08955*, 2020.