Florida International University - University of Miami TRECVID 2020 DSDI Track

Maria Presa-Reyes[1], Yudong Tao[2], Shu-Ching Chen[1], and Mei-Ling Shyu[2]

[1]School of Computing and Information Sciences
Florida International University, Miami, FL 33199, USA
[2]Department of Electrical and Computer Engineering
University of Miami, Coral Gables, FL 33146, USA
mpres029@cs.fiu.edu, yxt128@miami.edu,
chens@cs.fiu.edu, shyu@miami.edu

Abstract

This paper presents the framework and results from the team "Florida International University-University of Miami (FIU-UM)" in the TRECVID 2020 Disaster Scene Description and Indexing (DSDI) task. We submitted four runs, each applying the same framework but using different score aggregation methods to rank each video shot. The score aggregation methods used in these runs are summarized as follows.

- run1: the sum of the feature scores obtained from the video frames to rank the shots;
- run2: the average of the feature scores obtained from the video frames to rank the shots;
- run3: the maximum of the feature scores obtained from the video frames to rank the shots;
- run4: the average of the top three features' scores obtained from the video frames to rank the shots.

Our framework includes the following processing steps: (1) pre-processing imagery from the provided LADI (Low Altitude Disaster Imagery) dataset; (2) generating soft labels for imagery in the LADI dataset through the fusion of annotations from both human and machine annotators as well as image time/location-based concept lookups of open datasets; (3) categorizing the frames in the LADI imagery by five Convolutional Neural Network (CNN) models (i.e., damage, environment, infrastructure, vehicles, and water), each focused on a subset of the 32 features; and (4) aggregating the predictive scores of the frame-level to the shot-level through `sum`, `avg`, `max`, and `top`. To improve the performance of the CNN models, we adopt various training strategies, including (1) the model pre-trained on ImageNet; (2) propagating the labels during training, following the sequence nature of the LADI dataset; and (3) retrieving more relevant data using an image crawler to enhance the training data. This year, the FIU-UM team achieved the first place among all the submitted runs, regardless of the training type. Among a total of four prioritized submitted runs with different relevancy sorting techniques, three of our runs ranked the top 3. The submission details are listed as follows.

- Training type: LADI + Others (O)
- Team ID: FIU-UM (Florida International University - University of Miami)
- Year: 2020

I. Introduction

The TREC Video Retrieval Evaluation (TRECVID) [1] is a competition led by the National Institute of Standards and Technology (NIST), which aims to accelerate the research and development in video-based content analysis and retrieval [2]. The introduction of the Disaster Scene Description and Indexing (DSDI) track this year allowed our team to leverage our comprehensive knowledge and previous work in disaster data management [3–9] and our past experiences competing in other TRECVID tracks [10–12]. Among a total of four prioritized submitted runs with different relevancy sorting techniques, three of our run submissions ranked the top 3 among all the submissions.

The DSDI track gives the participants the access to the LADI (Low Altitude Disaster Imagery) dataset [13] to train their models. LADI is composed of imagery collected by the CAP (Civil Air Patrol) from a low-flying aircraft and hosted by the Federal Emergency Management Agency (FEMA). The dataset emphasizes unique disaster-related features such as the damage labels and scene descriptors. Image variations, including lighting, orientation, perspective, and resolution, are a key component to the LADI dataset. Any technology or tool developed to support disaster response will need to handle these types of variations. Convolutional Neural Network (CNN) has proven to generalize well when training images with variations while also achieving impressive results in the image recognition task [14].

The LADI data employs a hierarchical labeling scheme of five coarse categories and then more specific annotations for each category. Each image also contains valuable metadata with information on the camera used to take the photo and the aircraft's location and altitude. A subset of the LADI dataset, representing more than forty thousand images, were hand-annotated using the Amazon Mechanical Turk (MTurk) service. Moreover, the LADI dataset also includes machine-generated labels from commercial and open-source image recognition tools to provide additional context.

Delivering an effective response requires quick and precise analyses concerning the impact of a disastrous event. Data collected through remote sensing technologies, such as high-flying aircrafts or drones, have become crucial tools [15–19]

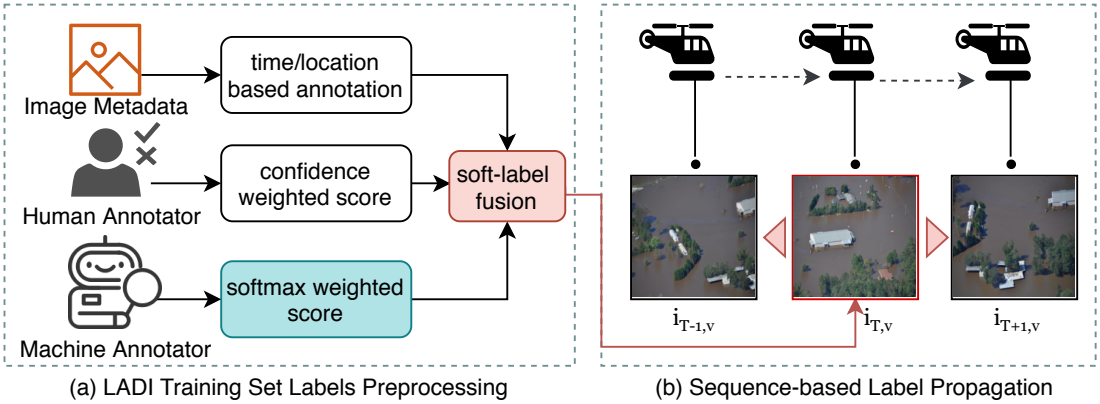

Fig. 1. LADI training set label processing and propagation (a) All information that can be obtained from the image metadata, human annotators, and machine annotators is fused; (b) More images are added to the training set

to survey the damage levels of the affected regions that became inaccessible due to the disastrous event. By leveraging the advanced technologies and machine learning methods such as deep learning [20, 21] during a disaster, it is possible to send a drone ahead of the search team to rapidly identify regions that are the most affected and should be prioritized. The automatic content-based analysis and classification of the features found in the recorded videos would provide the augmented curation and retrieval of the relevant information for situational awareness [22, 23].

One of the major challenges encountered when working with LADI was handling a large and mostly unlabeled dataset with a limited number of samples with noisy labels. Through some label-propagation, the support of open-source pre-trained models, and the available data sets from other sources, our team improved the LADI training labels significantly, making it possible to train robust models and obtain excellent results.

The crowdsourced human labels provided for a small subset of LADI imagery were highly imbalanced, with several images containing often incorrect labels. The soft-label assignment approach helps overcome such a problem. Soft-labels provide the model with knowledge in terms of the significance of each feature. Moreover, this approach works well given that we want to solve a ranking task. Through soft-labels, the model is trained to identify not only the occurrence of a certain feature within an image, but also how substantial this feature is. It also allowed us to better integrate the soft-labels from the human annotators and the SoftMax weights provided by the pre-trained models and the various commercial classifiers made available by the DSDI task coordinators.

We further implemented a label-propagation approach unique to the nature of the source for the training images. Imagery in LADI is taken following a sequence, much like a video [24–26]. Using the time and location metadata from the images, we generate that sequence and propagate the labels nearest to the image containing the highest ground truth soft-labels. If an image includes a particular feature, it is very likely that the image taken before or after includes the said feature as well. For better flexibility, five separate CNN models were trained for the features belonging to each coarse category (i.e., damage, environment, infrastructure, vehicles, and water).

During inference, the testing video shots are split into multiple image frames and fed to the five categorical models to extract the scores for each of the 32 features. Then four shot-level aggregations of the frame-level scores are implemented to rank the video shot according to its significance to support the content-based retrieval [27–29].

The remainder of this paper is structured as follows. Section II explains the proposed framework for the TRECVID 2020 DSDI task and the details of different strategies used in each run. Section III evaluates the performance of each submission and demonstrates the submission results. Section IV concludes the paper and suggests future directions for next year's submission.

II. The Proposed Framework

The proposed framework starts with the LADI training set preparation and pre-processing. Given that the DSDI track is a pilot challenge featuring a new dataset, a major focus has been put on exploring the data and finding ways to improve its label set by fusing different annotations from both human and machine labelers. As shown in Figure 1(a), annotations from different sources are integrated to create the soft-labels that allow us to train the models. The LADI feature set along with the annotations from other sources are described in Table I. These labels are then propagated throughout the LADI set, adding more curated images to the training set as we train the model.

TABLE I
A SUMMARY OF THE 32 FEATURES WITHIN THE 5 CATEGORIES AND SOME EXAMPLES OF ITS MATCHING MACHINE-BASED ANNOTATIONS AND METADATA CONCEPT LOOKUP

Category	**Features**	**Machine Annotation Concepts**	**Metadata Concept Lookup**
Damage	(1) damage (misc) (2) flooding / water damage (3) landslide (4) road washout (5) rubble / debris (6) smoke / fire	-	**FEMA Disaster Declarations**: hurricane, fires, flooding, ...
Environment	(7) dirt (8) grass (9) lava (10) rocks (11) sand (12) shrubs (13) snow/ice (14) trees	**Places365**: volcano, rock arch, beach, igloo, mountain snowy,...	**NOAA Climate**: snowfall events and location **FEMA Disaster Declarations**: volcanic eruptions
Infrastructure	(15) bridge (16) building (17) dam / levee (18) pipes (19) utility or power lines / electric towers (20) railway (21) wireless/radio communication towers (22) water tower (32) road	**Places365**: apartment building, water tower, bridge, highway,... **GCV**: building, city, pipe, railway, suburb, ...	**OpenStreetMap**: building, bridge, lanes, highway, ...
Vehicles	(23) aircraft (24) boat (25) car (26) truck	**COCO**: car, truck, boat, and aeroplane	-
Water	(27) flooding (28) lake / pond (29) ocean (30) puddle (31) river / stream	**Places365**: lake/natural, ocean, beach, river **GCV**: flood, floodplain, river, river island, water, sea, ...	**OpenStreetMap**: natural water, pond, reservoir, waterway, ...

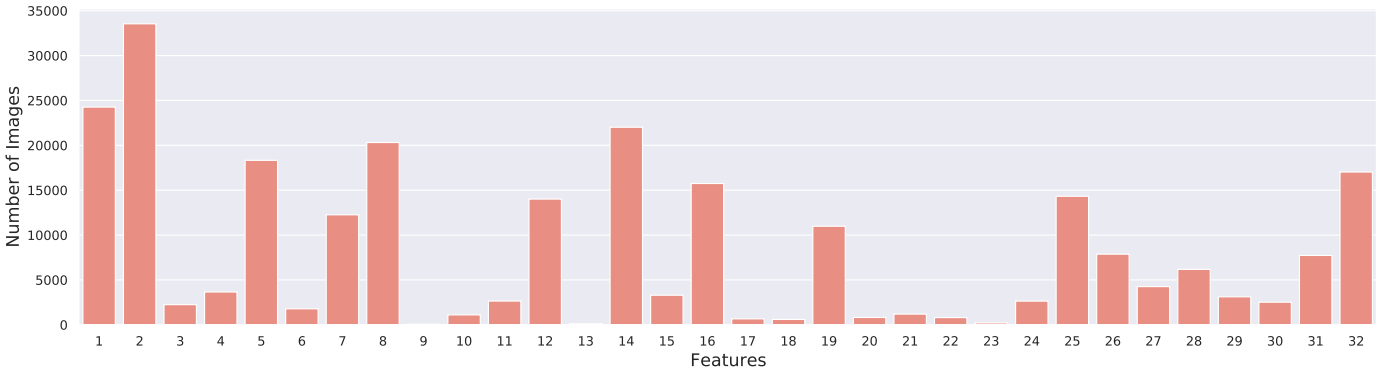

Fig. 2. The number of training images labeled by human annotators for each of the 32 features

A. Machine Annotators

LADI provides machine-generated labels for each image from some well-known open-source models and commercially available pre-trained models. These machine annotations generate each feature's label in the form of a numerical score indicating the relative confidence in the presence of the said feature. We further leveraged time and location metadata obtained from each image to include further concepts related to real-life events.

1) ResNet50 Pre-trained on Places365: Scene detection is included as one of the machine annotators and essential in improving the framework's performance. Among all the public scene detection datasets, Places365 incorporates 365 scene categories used to train the model [30]. A ResNet50 model trained on Places365 is applied to detect the location and environment in the LADI imagery. In the Places365 dataset, 1.8 million training images are provided, and each class includes at most 5000 images. This model provided many helpful concepts that enhanced the training set in terms of including images containing features under the categories for the environment, infrastructure, and water.

2) GCV: LADI provides machine-generated annotations from the commercially available pre-trained models marketed by Google Cloud Vision (GCV) [31]. GCV offers a number of products, of which LADI provides the scores for (1) the GCV *label detection* service and (2) the GCV *web entity detection*. The GCV API offers powerful pre-trained machine learning models

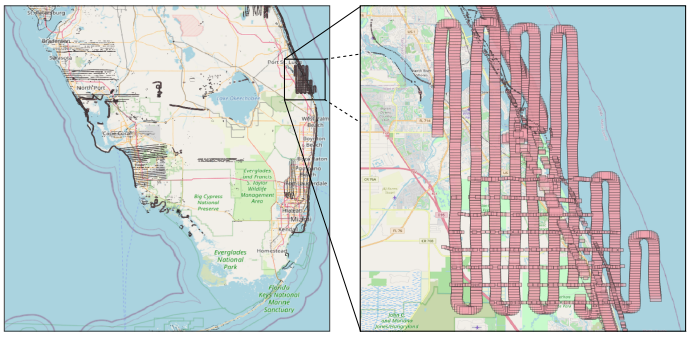

Fig. 3. LADI image footprints of the photos taken in South Florida from different flight missions.

to rapidly assign labels to images and quickly classify them into millions of predefined categories. The *web entity detection* detects web references to an image and returns a list of recommended tags.

3) YOLOv4 Pre-trained on COCO: Other than the previously described ResNet50 and GCV models, our team also applied the inference from the YOLOv4 model pre-trained on the COCO dataset [32]. YOLO (You Only Look Once) is a real-time object detection deep learning architecture proposed by Redmon *et al.* in 2015 [33]. YOLO trains on full images and directly optimizes detection performance while treating the detection mechanism as a regression problem. YOLO is fast compared to other detection networks. Microsoft COCO (Common Objects in Context) is large-scale object detection, segmentation, and captioning dataset. Moreover, the annotations provided by the YOLOv4 model trained on COCO include relevant features such as car and truck and have proven to be crucial in enhancing the model developed for the vehicle category.

B. Human Annotators

A subset of images from the LADI training set were annotated using Amazon Mechanical Turk (MTurk) [34]. The crowdsourced human labels provided for a subset of more than 40k LADI imagery were highly imbalanced as shown in Figure 2 with several images containing often incorrect labels. Such a problem was overcome through a soft label assignment approach. Each image is assigned a value from 0 to 1 for a specific feature, using the number of votes from each worker as a weight for the score. The more workers that assign a feature for a certain image, the higher the confidence in the image containing the correct feature. Each image i is first assigned a count $x_{i,C}$ indicating how many workers have labeled this image as a feature C. Given this information, the score is calculated as follows, where $x_C^{\texttt{min}}$ and $x_C^{\texttt{max}}$ are the minimum and maximum counts for a set of images in feature C.

$$score_{i,C} = \frac{x_{i,C} - x_C^{\texttt{min}}}{x_C^{\texttt{max}} - x_C^{\texttt{min}}} \tag{1}$$

C. Image Metadata Concept Lookup

LADI imagery follows the Exif (Exchangeable image file format) which has standard tags for valuable metadata. The information that can be extracted from the image metadata includes the focal length F, altitude A, latitude, longitude, camera model, and so on. This information can be utilized to estimate the geographical area covered by the photograph. Although there is no direct access to the height Sh and width Sw of the camera sensor through Exif, we acquired this information using the camera model provided by the metadata. Around 152 different camera models were identified from the images' metadata. Figure 3 demonstrates the polygons for the footprints of each image from different missions, which also shows how the image capture follows the sequence of the flight of the aircraft. The computed image area is only a rough estimate. To be able to calculate the exact geographical bounds that the image has covered, the angle from which the picture was taken is a required parameter [35]. The current footprint of a camera pointing straightly down is determined by some very basic trigonometry. Nonetheless, some drones in the market today already provide this type of information.

The estimated height and width of the image footprint can be calculated as follows.

$$\text{width} = \frac{A \times Sw}{F}, \text{height} = \frac{A \times Sh}{F} \tag{2}$$

The time and location metadata of the image provides some valuable clues in relation to the semantic content of the image [36, 37]. Given these two attributes, further information about the photographed area can be retrieved from open datasets to include more context considering the specific events, locations, and special weather conditions that might have been captured. FEMA Disaster Declarations is a notable source describing all federally declared disasters in the United States. This source is particularly valuable in the inclusion of images which features are under the damage category. While features of

other categories can be detected from models trained from a well-curated open dataset, damage assessment is still a very new research direction in the field of image classification. NOAA Climate has also been a good source to detect extreme weather events and occasions for snowfall. Other sources, such as OpenStreetMaps [38], provide a useful location-based context of the aerial images captured.

D. Soft-Label Fusion and Propagation

Soft-labels can provide the model with information about the significance of each feature in the image during training [39]. Before fusing the soft labels, a match is made at the semantic-level of the concept's name. Correlation studies [40] were conducted between the soft labels obtained from human annotations and the machine-generated scores. However, this approach did not perform well when working with very underrepresented features labeled by the human annotators, such as (9) lava and (23) aircraft (as shown in Figure 2). Hence, by matching machine and human annotations using the semantic similarity of their names, we obtain better results. If one image has been assigned two scores for the same feature from different annotators, then the highest score is kept.

We implemented a label-propagation approach following the time and location sequence data found on the LADI imagery's metadata. Labels from an image with a high feature score can be propagated to those images that are closest within the sequence. Label-propagation is done during the training of the model. Namely, if a model is shown to stop improving as it trains, it triggers an early-stop. The program will then expand the training set by propagating the score to the neighbors of the highest scored images, using the model being trained to assign the said scores.

As shown in Figure 1(b), scores are propagated within the neighbor images of the labeled image from video v. The notion is that if an image $i_{T,v}$, taken at a time T, contains a particular feature, it is very likely that the image taken before (i.e., $i_{T-1,v}$) and/or the image taken after (i.e., $i_{T+1,v}$) include the said feature. This concept is particularly true when working with the features under the "damage" category. For example, during a mission, crew members taking photographs from an aircraft after a disaster event are attentive in always pointing the camera towards the damages, capturing these critical features from different angles.

E. Categorical Model Setup and Training

Our deep learning framework is composed of five categorical models implemented based on the InceptionV3 architecture, with each model trained on the scores for the features of the specific category. Following the transfer learning approach [41], we fine-tune the weights of the entire network that has been pre-trained on ImageNet [14]. The last classification-head of the network is replaced by a dense layer implementing the sigmoid activation function for multi-class classification of soft-labels. During training, the binary crossentropy function calculates the model loss and updates the weights of the model accordingly. Adam solver is employed to optimize our model with a starting learning rate ($\eta = 1e-4$). The chosen learning rate is small enough to update the transferred weights slowly when fine-tuning the pre-trained model—achieving a more optimal set of final weights [42]. During training, the learning rate will drop to 10% of its current learning rate if there are no improvements to the validation loss value for a total of 10 consecutive training epochs.

F. Video Shot Inference and Ranking

Each video shot from the test-set is first split into a total of N number of frames, through one-frame per second. The frame set goes through the prediction of each trained categorical model, where the scores are assigned to the features within each category. Then, we implement four different shot-level score aggregation techniques described as follows.

1) `sum`*:* Adding up all the scores for every frame j under a certain feature or category C to allow the final score to consider all values assigned from every frame. Such an approach would bring to the top results of the video shots that have a longer footage containing the feature C.

$$\sum_{j=1}^{N} score_{j,C} \tag{3}$$

2) `avg`*:* By calculating the average of the scores, we apply a normalization effect to the sum of the scores. This approach allows a more fair comparison between video shots of different lengths.

$$\frac{\sum_{j=1}^{N} score_{j,C}}{N} \tag{4}$$

3) `max`*:* This approach searches for the maximum value of a score assigned under a feature C, and assigns this value as the score of the shot-level video.

$$\max_{j=1}^{N} score_{j,C} \tag{5}$$

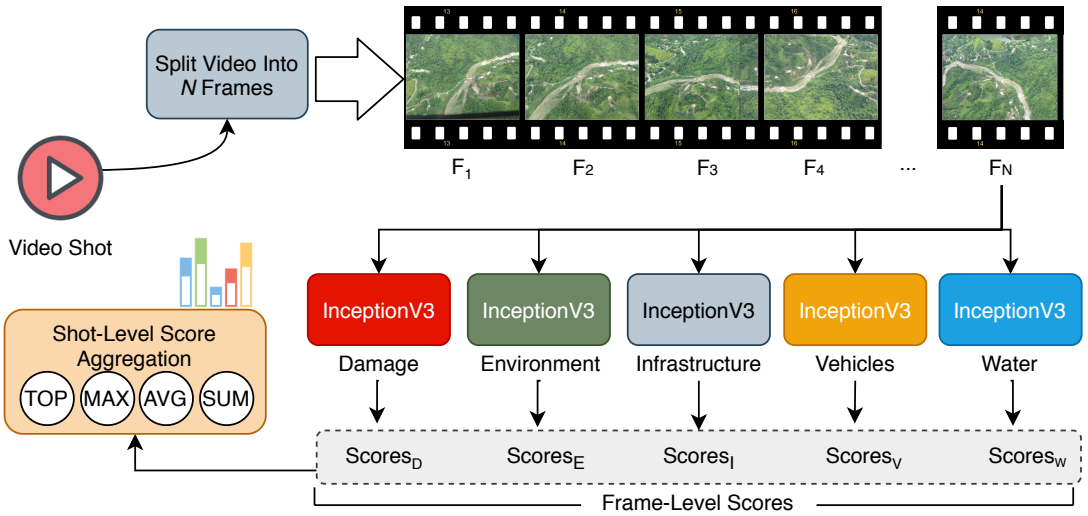

Fig. 4. Inference process starting with the split of the video shot into N frames and returning the aggregated scores of each feature

4) `top`*:* The average of the top-K scores is a similar approach to `avg`. The shot-level score is calculated as follows and $K = 3$ is used in our submitted run4.

$$\frac{\sum_{k=1}^{K} top_{k,C}}{K} \tag{6}$$

where $top_{k,C}$ refers to the k-th highest score among $\{score_{j,C}, j = 1, \ldots, N\}$.

G. Image Crawler

Even after the fusion of multiple annotations and time/location-based concept lookups, several features remained overwhelmingly underrepresented. We automatically crawl images using an image search engine, such as Bing Image[1] to enhance the training data, filter the outliers in the search engine results, and then train the classifier to detect these underrepresented features. Crawling for more data helps us overcome some of the imbalanced issues within the training set.

H. Submitted Runs

A total of four runs were submitted to the TRECVID 2020 DSDI task following the LADI + Others (O) training type. In all these runs, the five trained categorical CNN models based on InceptionV3 are used to generate the scores for each feature at the frame-level of each video shot, as shown on Figure 4.

The difference among all the submitted runs is the computation of the final score, namely, the shot-level score aggregation functions.

- **run1**: the aggregated sum of the scores obtained from the video frames to rank the shot;
- **run2**: the aggregated average of the scores obtained from the video frames to rank the shot;
- **run3**: the aggregated maximum obtained from the video frames to rank the shot;
- **run4**: the aggregated averages of top three features scores obtained from the video frames to rank the shot.

III. Results

A. Evaluation

Our proposed framework processes all the video shots in the test dataset and ranks them based on the predicted relevance to each feature of interest [28, 43]. For each of the given features, the top-1000 relevant video shots' IDs were submitted to be evaluated by the competition coordinators.

The test dataset for the DSDI track contains 1825 video shots with a total duration of around 5 hours, and each video shot is about maximum 60 seconds. They are collected from a recent natural disaster event operational videos. All the videos are evaluated by the assessors at NIST and annotated whether they are related to each feature of interest [44]. The Mean Average Precision (MAP) metric is computed to evaluate and compare the performance of different approaches.

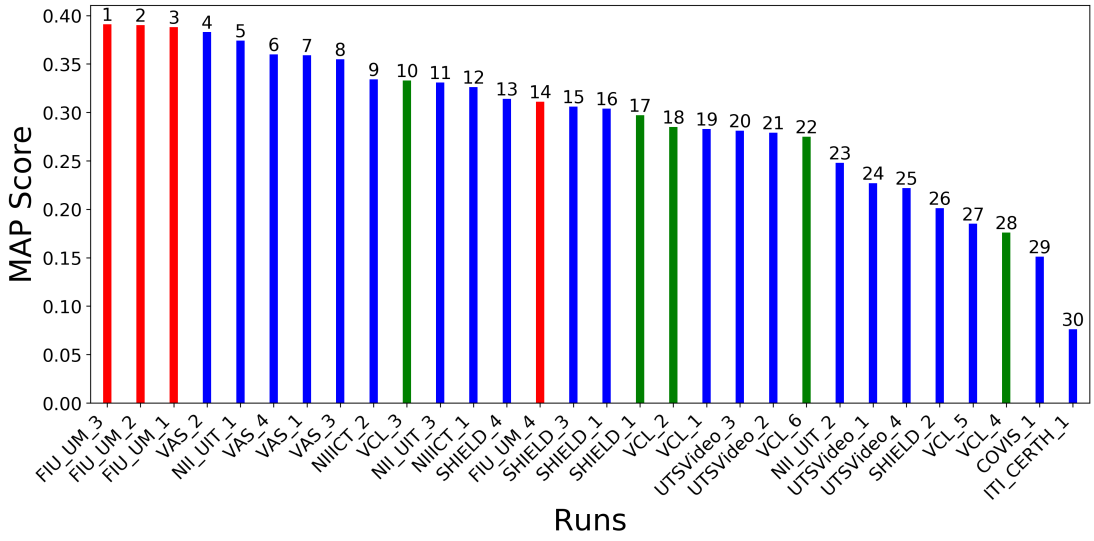

Fig. 5. Comparison of MAP scores among FIU_UM runs (red) with all the other submitted runs. Runs using only LADI-based data are labeled as blue and runs using LADI+Others data are labeled as green. All our runs use LADI+Others. The rank of each run is labeled on top of the corresponding bar.

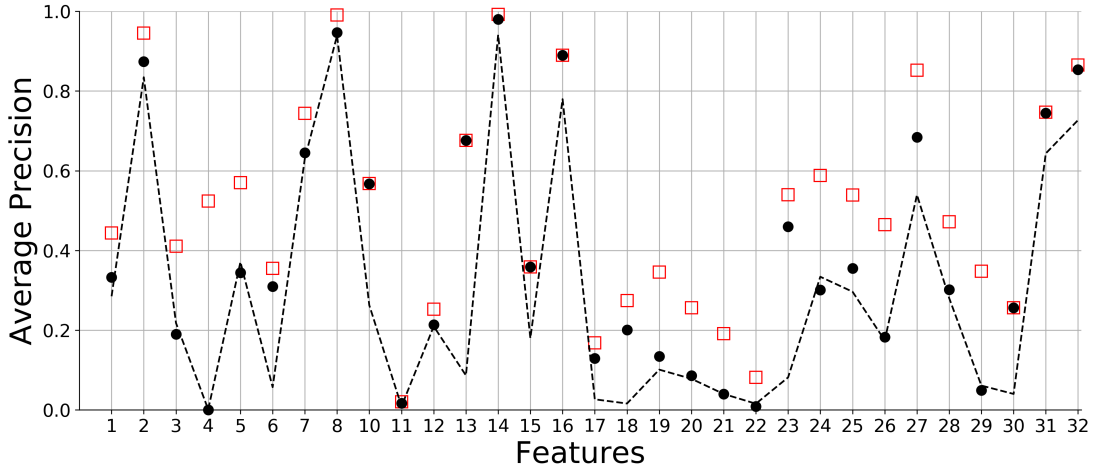

Fig. 6. Precision (dot) of our best run (run3), the median (dashes), and the best (box) results among all the submitted runs, regardless of the training type.

B. Performance

The MAP scores of the runs based on our proposed framework and all other submitted runs are shown in Figure 5. All our submitted runs utilize the data from LADI and other datasets, as mentioned above in the training phase. Their MAP scores are 0.388 (run1), 0.390 (run2), 0.391 (run3), and 0.311 (run4), which ranked 3rd, 2nd, 1st, and 14th among all the submitted runs, respectively. The best ranking submission run makes use of the `max` shot-level score aggregation approach. The great performance for `max` compared to our other submission runs demonstrates how the noise found in some of the frames may have undermined the overall performance.

Figure 6 shows the mean average precision of each feature of our best run (run3). The x-axis of Figure 6 shows the feature number, while the y-axis presents the average precision measures of our run (shown as a dot), median performance among all the submitted runs (shown as dashes), and the best-performed run (shown as a box) for each feature. These feature-level metrics indicate that we perform the best in features 13 (snow/ice), 15 (bridge), 16 (building), and 30 (puddle). Also, the best performed average precision scores of features 6 (smoke/fire), 10 (rocks), and 31 (river/stream) are achieved by the other runs submitted by us. These good results are achieved by virtue of different parts of our proposed framework. For instance, the annotations provided by both GCV and Places365 and some location-based geographical features captured from

[1]https://www.microsoft.com/en-us/bing/apis/bing-image-search-api

OpenStreetMaps are the major contributors to the great performance achieved by feature 16 (building). Moreover, the proposed concept lookup of the image metadata with the time and location of real-life events and weather conditions proves to work well for the event-based features such as 6 (smoke/fire) and 13 (snow/ice). One of the weaknesses of the proposed framework is the noise and mislabeling found on the data annotations. We demonstrated many approaches taken to overcome this challenge through the introduction of soft labels and label propagation. Further processing and curation of the data labels will guarantee better improvements.

IV. Conclusion and Future Work

In this notebook paper, the framework and results of the FIU-UM team in the TRECVID 2020 DSDI task are presented. This track's introduction this year allowed the FIU-UM team to leverage our comprehensive knowledge and previous work in disaster data management and valuable past experiences competing in other TRECVID tracks. Because the DSDI track is a pilot challenge this year featuring a new dataset, the major focus has been put to explore the data and improve the label-set. As part of our future work, we will enhance the proposed framework by developing one single model that supports the hierarchical labeling style of the LADI dataset. Moreover, we will explore ways to also consider the sequence information of the images to further improve the model performance.

V. Acknowledgements

We would like to thank Mario Jacas and Rui Ma, for their valuable technical support on this project.

References

[1] G. Awad, A. A. Butt, K. Curtis, Y. Lee, J. Fiscus, A. Godil, A. Delgado, J. Zhang, E. Godard, L. Diduch, J. Liu, A. F. Smeaton, Y. Graham, G. J. F. Jones, W. Kraaij, and G. Quénot, "TRECVID 2020: comprehensive campaign for evaluating video retrieval tasks across multiple application domains," in *TRECVID*. NIST, USA, 2020.

[2] S.-C. Chen, R. L. Kashyap, and A. Ghafoor, *Semantic models for multimedia database searching and browsing*. Springer Science & Business Media, 2006, vol. 21.

[3] S. Pouyanfar, Y. Tao, H. Tian, S.-C. Chen, and M.-L. Shyu, "Multimodal deep learning based on multiple correspondence analysis for disaster management," *World Wide Web*, vol. 22, no. 5, pp. 1893–1911, 2019.

[4] T. Wang, Y. Tao, S.-C. Chen, and M.-L. Shyu, "Multi-task multimodal learning for disaster situation assessment," in *IEEE Conference on Multimedia Information Processing and Retrieval*. IEEE, 2020, pp. 209–212.

[5] H. Tian, H. C. Zheng, and S.-C. Chen, "Sequential deep learning for disaster-related video classification," in *IEEE Conference on Multimedia Information Processing and Retrieval*. IEEE, 2018, pp. 106–111.

[6] M. E. P. Reyes, S. Pouyanfar, H. C. Zheng, H.-Y. Ha, and S.-C. Chen, "Multimedia data management for disaster situation awareness," in *International Symposium on Sensor Networks, Systems and Security*. Springer, 2017, pp. 137–146.

[7] H. Tian, S.-C. Chen, S. H. Rubin, and W. K. Grefe, "FA-MCADF: Feature affinity based multiple correspondence analysis and decision fusion framework for disaster information management," in *IEEE International Conference on Information Reuse and Integration*. IEEE, 2017, pp. 198–206.

[8] L. Zheng, C. Shen, L. Tang, T. Li, S. Luis, S.-C. Chen, and V. Hristidis, "Using data mining techniques to address critical information exchange needs in disaster affected public-private networks," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2010, pp. 125–134.

[9] L. Zheng, C. Shen, L. Tang, T. Li, S. Luis, and S.-C. Chen, "Applying data mining techniques to address disaster information management challenges on mobile devices," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2011, pp. 283–291.

[10] Y. Yan, S. Pouyanfar, Y. Tao, H. Tian, M. Reyes, M. Shyu, S. Chen, W. Chen, T. Chen, and J. Chen, "FIU-UM at TRECVID 2017: Rectified linear score normalization and weighted integration for ad-hoc video search," in *TRECVID*. NIST, USA, 2017.

[11] S. Pouyanfar, Y. Tao, H. Tian, M. E. P. Reyes, Y. Tu, Y. Yan, T. Wang, Y. Li, S. Sadiq, M.-L. Shyu, S.-C. Chen, W. Chen, T. Chen, and J. Chen, "Florida International University-University of Miami TRECVID 2018," in *TRECVID*. NIST, USA, 2018.

[12] Y. Tao, T. Wang, D. Machado, R. Garcia, Y. Tu, M. P. Reyes, Y. Chen, H. Tian, M.-L. Shyu, and S.-C. Chen, "Florida International University-University of Miami TRECVID 2019," in *TRECVID*. NIST, USA, 2019.

[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*, Sep. 2019, pp. 1–6.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[15] J. Yan, K. Zhang, C. Zhang, S.-C. Chen, and G. Narasimhan, "Automatic construction of 3-D building model from airborne lidar data through 2-d snake algorithm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 1, pp. 3–14, 2014.

[16] K. Zhang, J. Yan, and S.-C. Chen, "Automatic construction of building footprints from airborne LIDAR data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 9, pp. 2523–2533, 2006.

[17] K. Zhang, S.-C. Chen, P. Singh, K. Saleem, and N. Zhao, "A 3D visualization system for hurricane storm-surge flooding," *IEEE Computer Graphics and Applications*, vol. 26, no. 1, pp. 18–25, 2006.

[18] K. Zhang, S.-C. Chen, D. Whitman, M.-L. Shyu, J. Yan, and C. Zhang, "A progressive morphological filter for removing nonground measurements from airborne lidar data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 4, pp. 872–882, 2003.

[19] M. Presa-Reyes and S.-C. Chen, "Assessing building damage by learning the deep feature correspondence of before and after aerial images," in *IEEE Conference on Multimedia Information Processing and Retrieval*. IEEE, 2020, pp. 43–48.

[20] Y. Yan, M. Chen, M.-L. Shyu, and S.-C. Chen, "Deep learning for imbalanced multimedia data classification," in *IEEE International Symposium on Multimedia*. IEEE, 2015, pp. 483–488.

[21] S. Pouyanfar, S. Sadiq, Y. Yan, H. Tian, Y. Tao, M. E. P. Reyes, M. Shyu, S. Chen, and S. S. Iyengar, "A survey on deep learning: Algorithms, techniques, and applications," *ACM Computing Surveys*, vol. 51, no. 5, pp. 92:1–92:36, 2019.

[22] S.-C. Chen, S. Sista, M.-L. Shyu, and R. L. Kashyap, "Augmented transition networks as video browsing models for multimedia databases and multimedia information systems," in *IEEE International Conference on Tools with Artificial Intelligence*, 1999, pp. 175–182.

[23] X. Li, S.-C. Chen, M.-L. Shyu, and B. Furht, "Image retrieval by color, texture, and spatial information," in *International Conference on Distributed Multimedia System*, 2002, pp. 152–159.

[24] L. Lin and M.-L. Shyu, "Weighted association rule mining for video semantic detection," *International Journal of Multimedia Data Engineering and Management*, vol. 1, no. 1, pp. 37–54, 2010.

[25] S.-C. Chen, M.-L. Shyu, and C. Zhang, "Innovative shot boundary detection for video indexing," in *Video Data Management and Information Retrieval*. IGI Global, 2005, pp. 217–236.

[26] S.-C. Chen, M.-L. Shyu, C. Zhang, and R. L. Kashyap, "Video scene change detection method using unsupervised segmentation and object tracking." in *IEEE International Conference on Multimedia & Expo*, 2001.

[27] T. Meng and M.-L. Shyu, "Leveraging concept association network for multimedia rare concept mining and retrieval," in *IEEE International Conference on Multimedia and Expo*, July 2012, pp. 860–865.

[28] M.-L. Shyu, S.-C. Chen, M. Chen, and C. Zhang, "A unified framework for image database clustering and content-based retrieval," in *ACM International Workshop on Multimedia Databases*, 2004, pp. 19–27.

[29] X. Huang, S. Chen, M. Shyu, and C. Zhang, "User concept pattern discovery using relevance feedback and multiple instance learning for content-based image retrieval," in *International Workshop on Multimedia Data Mining*, 2002, pp. 100–108.

[30] 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*, vol. 40, no. 6, pp. 1452–1464, 2017.

[31] D. Mulfari, A. Celesti, M. Fazio, M. Villari, and A. Puliafito, "Using google cloud vision in assistive technology scenarios," in *IEEE Symposium on Computers and Communication*. IEEE, 2016, pp. 214–219.

[32] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *European Conference on Computer Vision*. Springer, 2014, pp. 740–755.

[33] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.

[34] A. M. Turk, "Amazon mechanical turk," *Retrieved August*, vol. 17, 2012.

[35] R. Augustine, "Aerial camera ground footprint with a gimbal," Jan 2018. [Online]. Available: http://rijesha.com/blog/aerial-cam-footprint/

[36] S.-C. Chen and R. L. Kashyap, "Temporal and spatial semantic models for multimedia presentations," in *International Symposium on Multimedia Information Processing*, 1997, pp. 441–446.

[37] D. Liu, Y. Yan, M.-L. Shyu, G. Zhao, and M. Chen, "Spatio-temporal analysis for human action detection and recognition in uncontrolled environments," *International Journal of Multimedia Data Engineering and Management*, vol. 6, no. 1, pp. 1–18, 2015.

[38] C. Bansal, A. Singla, A. K. Singh, H. O. Ahlawat, M. Jain, P. Singh, P. Kumar, R. Saha, S. Taparia, S. Yadav *et al.*, "Characterizing the evolution of indian cities using satellite imagery and open street maps," in *ACM SIGCAS Conference on Computing and Sustainable Societies*, 2020, pp. 87–96.

[39] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.

[40] Q. Zhu, L. Lin, M.-L. Shyu, and S.-C. Chen, "Feature selection using correlation and reliability based scoring metric for video semantic detection," in *IEEE International Conference on Semantic Computing*, 2010, pp. 462–469.

[41] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[42] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1, no. 2.

[43] X. Chen, C. Zhang, S.-C. Chen, and M. Chen, "A latent semantic indexing based method for solving multiple instance learning problem in region-based image retrieval," in *IEEE International Symposium on Multimedia*, 2005, pp. 37–44.

[44] A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and TRECVid," in *ACM International Workshop on Multimedia Information Retrieval*. ACM Press, 2006, pp. 321–330.