University of Missouri-Kansas City TRECVID 2022 DSDI Track

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

[1]Department of Electrical and Computer Engineering
University of Miami, Coral Gables, FL 33146, USA
[2]School of Science and Engineering
University of Missouri-Kansas City, Kansas City, MO 64110, USA
yxt128@miami.edu, s.chen@umkc.edu, shyum@umkc.edu

Abstract

This paper presents the framework and results from the team "University of Missouri-Kansas City (UMKC)" in the TRECVID 2022 Disaster Scene Description and Indexing (DSDI) task. This year our team submitted a total of two runs, one for the LADI-based submission run and the other for the LADI + Others Submission Run. The submitted results were obtained based on the following processing steps proposed in our submissions for TRECVID 2022 DSDI task: (1) pre-processing the imagery from the provided LADI (Low Altitude Disaster Imagery) dataset; (2) generating soft labels for imagery in the LADI dataset through the advanced fusion of the annotations obtained from both human and machine annotators; and (3) categorizing the frames in the LADI imagery by the pre-trained Convolutional Neural Network (CNN) models, each focused on a different aspect of the target features. We use a variety of training strategies to improve the performance of the CNN models, including using a Confident Learning approach to denoise the training set and fusing the information from multiple models pre-trained on the well-known public dataset benchmarks. The final score is produced by (1) determining which characteristics from multiple models are semantically relevant to the target features in DSDI and (2) searching for the optimum approach to combine the predicted feature scores from multiple pre-trained models using a Differential Evolution optimization technique. The test video clips are then ranked according to their final feature scores that determine their relevance to a certain target feature. The UMKC team submitted two runs this year. The submission details are as follows.

- Training type: LADI-based (L) and LADI + Others (O)
- Team ID: UMKC (University of Missouri-Kansas City)
- Year: 2022

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 [3–8] and our past experiences competing in other TRECVID tracks [9–12]. In this year, we submitted two runs based on the framework proposed in TRECVID DSDI 2021, one for the L training type and the other for the O training type.

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 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 Networks (CNNs) have proven to generalize well when the training images are with variations while also achieving impressive results in the image recognition task [14].

The LADI dataset 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, was 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. The multimedia data, such as the one that can be found in LADI and in contrast to conventional data that consists of just texts and numbers, is often unstructured and noisy. Conventional data analysis will not be able to handle this massive volume of complex data. Hence, more extensive and sophisticated solutions are necessary [15–19].

Delivering an efficient response requires a timely and accurate analysis of the impacts of a disaster. Data obtained using remote sensing technology, such as high-flying aircrafts or drones, has proven to be critical in assessing the extent of damage in areas that have been inaccessible as a result of the disaster [20–24]. By leveraging the advanced technologies and machine learning methods such as deep learning [25–27] 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 [28–32].

One of the major challenges encountered when working with LADI was handling a large and mostly unlabeled dataset with a limited number of samples that at best included some noisy labels. This posed a substantial challenge in developing an appropriate catastrophe scene description model. Furthermore, the crowd-sourced human annotations supplied for a section of the LADI training set are very imbalanced and untrustworthy, with some photos including mislabels on a regular basis. We further enhanced the LADI training labels with the help of open-source pre-trained models and datasets from multiple sources, allowing us to reach an exceptional performance.

Considering the mislabels that can be found in LADI's labeled subset, the soft-label assignment method aids in the solution of such a challenge. Soft labels offer information to the model about the relevance of each target feature. The model is trained to recognize the existence of a given feature inside an image and how significant that feature is, using the soft labels. Such an approach has shown to be very effective in addressing the ranking issue. It also enables us to better combine the soft labels supplied by human annotators, the SoftMax weights provided by the pre-trained models, and the numerous commercial classifiers made accessible by the DSDI task coordinators.

In TRECVID2022-DSDI, our team adopted various training strategies, including (1) using the model pre-trained on ImageNet; (2) propagating the labels during training, following the sequence nature of the LADI dataset; and (3) optimizing the fusion weights based on the testing data in the previous years of TRECVID DSDI challenges [12, 33]. Imagery in LADI is taken following a sequence, much like a video [34]. Using the time and location metadata from the images, we generated that sequence and propagated 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 also 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). This year, our team utilizes the framework proposed in TRECDIV DSDI 2021 and tunes the model with testing data and labels in both 2020 and 2021 to obtain the optimized weights of each feature.

For inference, the testing video images are divided into numerous picture frames, which are then fed into the feature-score models to predict the scores for the 32 characteristics. In the next step, the feature-score fusion and aggregation of the frame-level scores are applied in order to rank the video shot according to its relevance to enable the content-based retrieval system [35, 36].

The remainder of this paper is structured as follows. Section II explains the proposed framework for the TRECVID 2022 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

For TRECVID DSDI 2022, our proposed framework inherits the framework used in TRECVID DSDI 2021 [37] based on confidence learning (CL) for weakly-supervised learning and differential evolution (DE) for feature fusion. As shown in Figure 1, we first utilize CL method to train a model on samples with better label quality rather than their quantity. Our CL-based method is conducted by leveraging a five-fold cross-validation to generate out-of-sample predicted probabilities for the training set, resulting in soft labels that can be used to train a model with confidence. Soft labels provide the advantage of allowing a model to be trained on the reliance of each target feature, alleviating some of the problems associated with highly imbalanced and noisy labels. Then, a collection of pre-trained models, which are trained on public data benchmarks such as ImageNet, is acquired and the feature scores are generated for the data. In the end, a final score that incorporates all of the models' relevant predictions into a single scalar is generated to rank the video clip, where the weights of the relevant predictions are obtained and optimized using DE. In the following, a brief summary of each component in the proposed framework is described.

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. Our team further includes other machine-generated annotations from open source pre-trained models only for the LADI + Others (O) type of submissions.

1) Inception-ResNet-V2 Pre-trained on ImageNet: ImageNet [14] is a well-known large-scale picture dataset including concepts from a variety of fields, such as animal, instrumentation, scene, and activity, all of which occur in some of the queries. ImageNet has 1.2 million photos in total, divided into 1000 classifications. This dataset contains a large number of real-world items, and the classification accuracy of those models trained on it has outperformed human performance using contemporary deep neural networks. To obtain the prediction scores for concepts in each keyframe from the final dense layer, we employ an Inception-ResNet-V2 [38] model pre-trained on the ImageNet dataset.

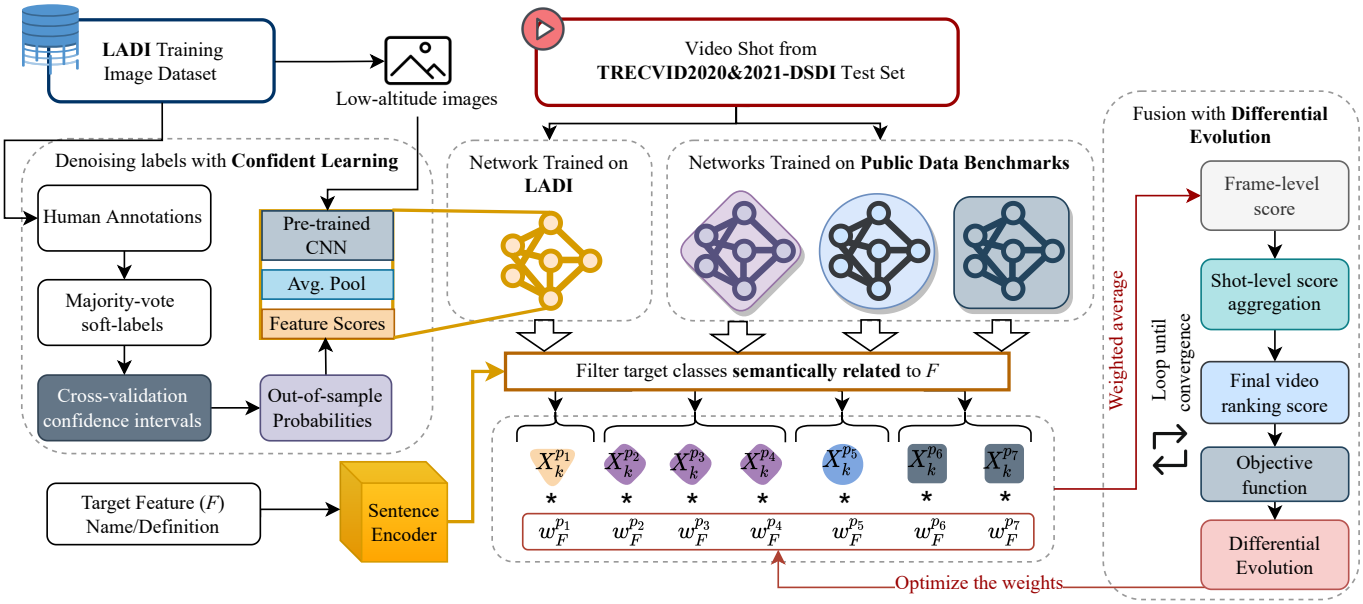


Fig. 1: The proposed weakly-supervised deep learning framework using a confident learning approach to denoise crowd-sourced annotations and a multi-modality fusion framework to search and combine relevant target features predicted by multiple pre-trained neural networks. Adapted from Figure 1 in [37].

2) 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 [39]. 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.

3) Google Cloud Vision: LADI provides machine-generated annotations from the commercially available pre-trained models marketed by Google Cloud Vision (GCV) [40]. 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 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.

4) 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 [41]. YOLO (You Only Look Once) is a real-time object detection deep learning architecture proposed by Redmon *et al.* in 2015 [42]. YOLO trains on full images and directly optimizes the 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 a 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.

5) ViT-B/16 Pre-trained on ImageNet21K: Alexey Dosovitskiy proposed the Vision Transformer (ViT) model [43] as a competitive alternative to CNNs. ViT is recently extensively employed in various image identification applications. The vision transformer model employs multi-head self-attention without the need for image-specific biases. The model divides the pictures into a series of positional embedding patches, which the transformer encoder processes. It does so in order to comprehend the image's local and global characteristics. Moreover, the ViT has been shown to achieve a greater accuracy rate with less training time on a big dataset than a regular CNN model. As part of our machine annotators, ViT-B/16 pre-trained on ImageNet21K plays a key part in our developed framework, especially in detecting smaller objects, such as the Utility-line, Car, Boat, and Truck.

6) ResNet50 Pre-trained on Incidents Dataset: The large-scale Incidents Dataset includes 446,684 scene-centric class-positive photos (annotated by humans) relating to natural catastrophes and forms of damage such as events that may need human attention or aid. The 43 categories covered by the Incidents Dataset are referred to as occurrences. A total of 49 distinct places were used to provide variations in the images. The dataset also includes 697,464 class-negative pictures, which were utilized for training the final model to reduce false-positive predictions.

B. Human Annotators

A subset of images from the LADI training set was annotated using Amazon Mechanical Turk (MTurk) [44]. The crowd-sourced human labels provided for a subset of more than 40k LADI imagery were highly imbalanced with several images containing often incorrect labels. Such a problem was overcome through a soft label assignment approach. Each Human Intelligence Task (HIT) on the MTurk platform, according to the LADI creators, asks the human worker whether any of the labels in each of the coarse categories are correct. Each HIT only asks about one category at a time. As a result, each HIT is given to three individuals in order to establish an agreement on label quality. If more validation was necessary, the HIT was outsourced to two more employees, bringing the total to five workers per category and picture. Each image is assigned a value from 0 to 1 for a specific target 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.

C. Confident Learning

The crowd-sourced annotations are used to give a value between 0 and 1 to each picture in our training set for a certain target feature. We use a CL technique to train a model with confidence on samples with a high predicted probability for their training labels, concentrating on label quality rather than quantity. Our CL-based technique begins with five-fold cross-validation to provide out-of-sample predicted probabilities for the training set, yielding soft labels that can be used to confidently train a model. Soft labels have the benefit of enabling a model to be trained on the dependency of each target feature, which eliminates some of the issues that come with extremely imbalanced and noisy labels. The soft labeling approach also correlates nicely with the DSDI track's goal of providing a continuous confidence measure, which performs favorably in resolving the ranking issue.

D. Feature Score Model Setup and Training

Our feature score model is trained on the LADI's confident labels generated by the CL-based technique and is based on the EfficientNet-B5 architecture [45]. Following the transfer learning approach [46], 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 the multi-class classification of soft labels. During training, the binary cross-entropy 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 [47]. 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.

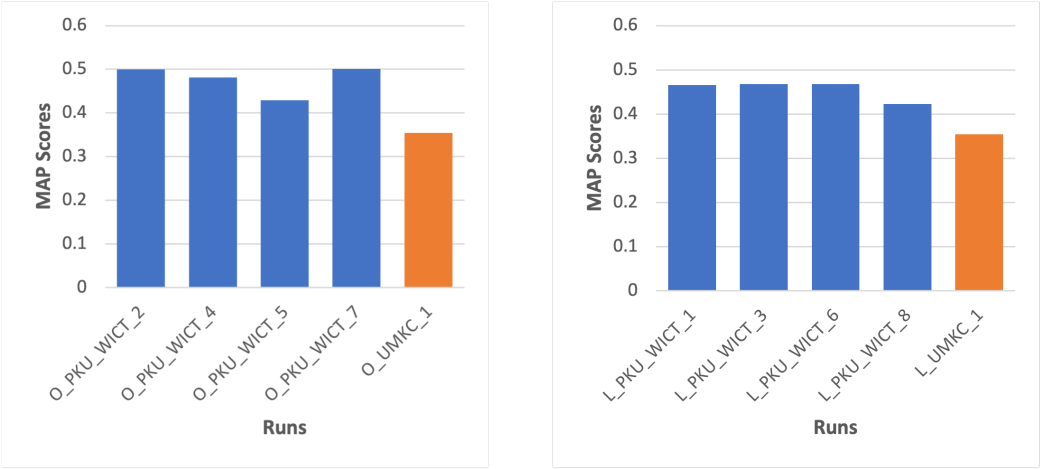
E. Feature Fusion

1) Target Feature Match: A semantic match of the feature's name (or definition) in both LADI and the pre-trained model's feature list is formed before the actual fusion of the scores. Semantic similarity uses Natural Language Processing (NLP) methods like word embedding to detect how similar two pieces of text are by their meaning. Text describing the target feature is encoded into high-dimensional vectors using the Universal Sentence Encoder, which may be used for text classification, semantic similarity, clustering, and other natural language applications. The Universal Sentence Encoder [48] has been pre-trained and is freely accessible on Tensorflow-hub. The encoder used in this work is based on the Deep Averaging Network (DAN), which averages the input embeddings for words and bi-grams before passing them through a feedforward deep neural network (DNN) to effectively construct the sentence embeddings. Moreover, this encoder uses unsupervised training data from various online sources, including Wikipedia, web news, web question-and-answer sites, and discussion forums. We then compute the cosine similarity of the two sentence embeddings to find a match given certain cosine distance thresholds.

2) Optimizing the Weights of the Pre-trained Models Using Differential Evolution: The weighted sum approach merges all model's predictions into a single scalar that can serve as a score to rank the video clip according to a target feature. The problem emerges while assigning the weighting coefficients since the answer is heavily dependent on the weighting factors selected [49]. This strategy of optimizing the following problem by constantly constructing a possible solution based on an evolutionary process is known as DE [50].

$$\hat{w}_F = \arg\min_{w_F} G(w_F) = \arg\min_{w_F} \left[1 - AP^N(V_F)\right] \tag{1}$$

where $AP^N(V_F)$ is the average precision score obtained for test data used in TRECVID DSDI 2020 and TRECVID DSDI 2021 challenges, and w_F refer to the weights of each selected feature. In order to rank the test video clips based on their relevance to a target feature, the final score is calculated by identifying the best possible combination of many scores received from a variety of models, and it is then utilized to calculate the final score.



(a) LADI + Others (O) Submission Runs

(b) LADI-based (L) Submission Runs

Fig. 2: Comparison of MAP scores among FIU-UM runs (orange) with all the other submitted runs in DSDI.

F. Submitted Runs

A total of two runs were submitted to the TRECVID 2022 DSDI task, where one following the LADI (L) training type and the other following the LADI + Others (O) training type. For the L training type run, only the LADI dataset that has been provided will be utilized in the development of our system. Both runs are built based on fully automated feature score fusion through differential evolution and based on the test data. The difference between the L and O runs is that the features used in the differential evolution algorithm are developed only based on the LADI dataset for the L run and, based on additional datasets for the O run.

III. Results

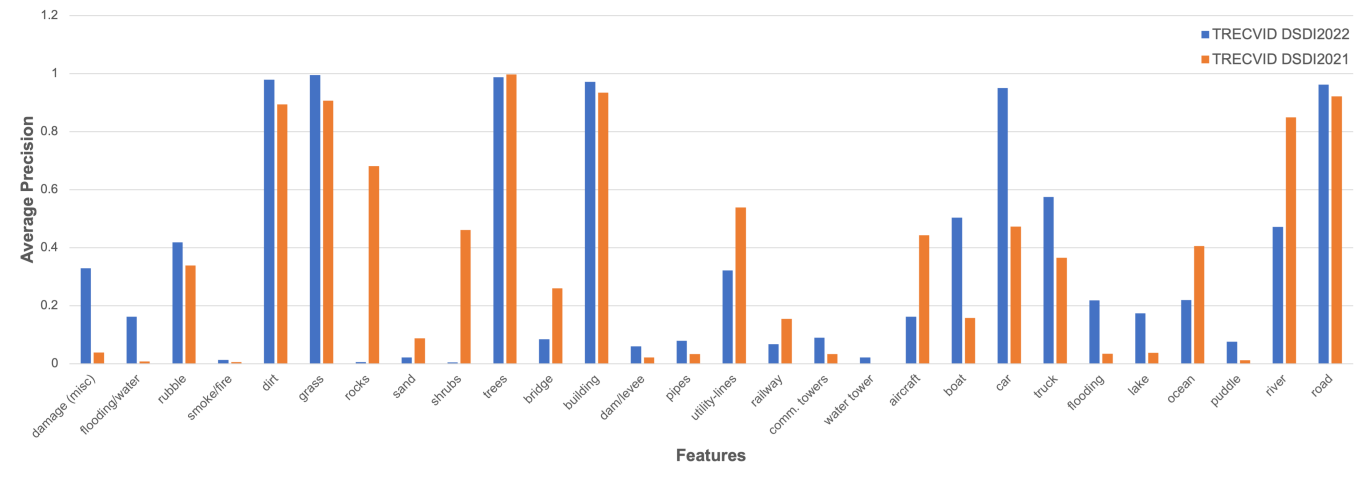
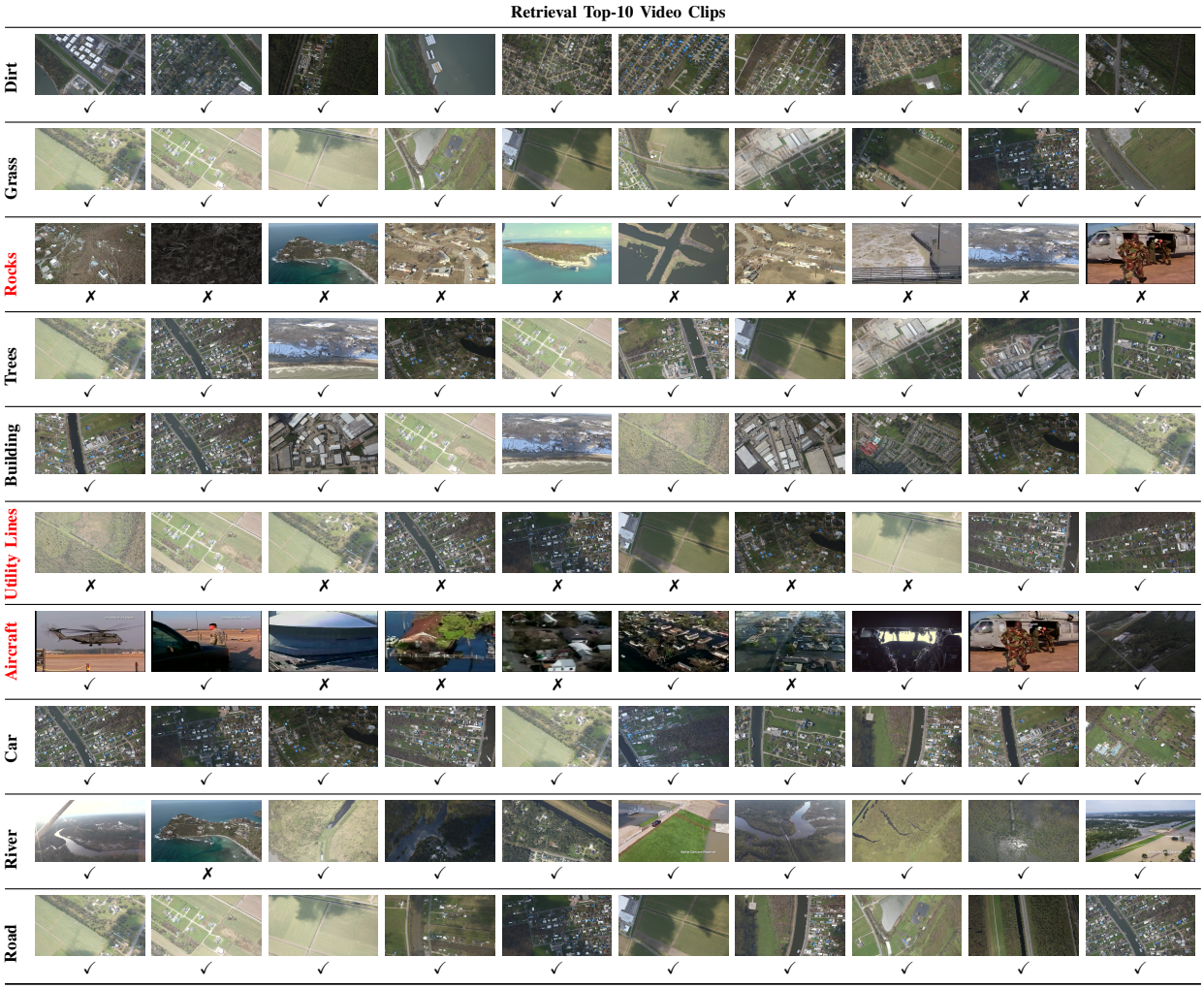


Fig. 3: The comparison of precision scores of a feature between the O run submitted to TRECVID2022-DSDI (O_UMKC_1) and the best O run submitted to TRECVID2021-DSDI (O_FIU_UM_1) using the same methodology.

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 [36]. 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 2,157 video shots. The videos were compiled from operational footage from previous natural catastrophe events. All the videos are evaluated by the assessors at NIST and annotations are generated to determine whether they are related to each feature of interest [51]. The Mean Average Precision (MAP) metric is computed to evaluate and compare the performance of different approaches.

TABLE I: Qualitative results of the first 10 video clips retrieved for selected features using our submitted solution O_UMKC_1. The features in red are select ones with significantly degraded performance compared to last year's submission.



B. *Performance*

The MAP scores of the runs based on our proposed framework and all other submitted runs are shown in Figure 2. Their MAP scores are both 0.354. This year's submissions both make use of the fully automated feature score fusion technique using DE proposed in [37]. Different from the last year's submission, we optimize the weights of feature fusion with both annotations for TRECVID DSDI 2020 and 2021 Test datasets.

Figure 3 summarizes the mean average precision (MAP) per target feature obtained by the best O runs for both this year and last year. The x-axis of the figure shows the DSDI target feature name, while the y-axis presents the average precision measure of each target feature. From the comparison between years, we can observe that some features (e.g., dirt, grass, trees, building, and roads) achieve similar performance while the performance of other features changes dramatically. These robustly performed features all describe common concepts that occur frequently in the LADI training dataset and other datasets used to generate other pre-trained models. However, for those less commonly seen features, especially those with largely different visual characteristics in different countries and regions (e.g., rocks, utility lines, etc.), the performance changes a lot over two years although the applied pipeline is the same. This potentially implies that the model's generalization capability toward a reliable and robust detection of these uncommon features remains a challenging problem.

Table I qualitatively summarizes the top 10 video clips retrieved for ten of our selected target features. This qualitative visual is meant to understand the limitations and achievements of our proposed method. For features that perform much worse than last year, it seems that the model is misled by the weight optimization process using last year's annotations given the changes in visual characteristics across various regions.

IV. Conclusion and Future Work

In this notebook paper, the framework and results of the UMKC team in the TRECVID 2022 DSDI task are presented. This year, we applied the same technique as proposed in TRECVID 2021 DSDI based on the Confident Learning (CL) strategy to build a model that could handle the noisy labels in the training set. The final score is determined by (1) evaluating which features from multiple models are semantically relevant to the DSDI target features and (2) using a method known as DE to find the optimum approach to combine the matching predicted scores from these models. The test video clips are then ranked according to their relevance to a particular feature in the final score.

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. Meanwhile, the model's performance on successive two years has been analyzed and we can observe that for uncommon features with different visual characteristics such as utility lines and electricity towers, the performance of the models changed quite significantly. Further research to improve the reliability and robustness of the proposed model should be conducted to improve the usability of the proposed model in real-world applications.

V. Acknowledgements

For Shu-Ching Chen, this research is partially supported by NSF CNS-1952089 and CNS-2125165.

References

[1] G. Awad, K. Curtis, A. A. Butt, J. Fiscus, A. Godil, Y. Lee, A. Delgado, J. Zhang, E. Godard, B. Chocot, L. Diduch, J. Liu, Y. Graham, , and G. Quénot, "An overview on the evaluated video retrieval tasks at trecvid 2022," in *Proceedings of TRECVID 2022*. NIST, USA, 2022.

[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] 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.

[5] 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.

[6] 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.

[7] 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.

[8] T. Li, N. Xie, C. Zeng, W. Zhou, L. Zheng, Y. Jiang, Y. Yang, H.-Y. Ha, W. Xue, Y. Huang *et al.*, "Data-driven techniques in disaster information management," *ACM Computing Surveys*, vol. 50, no. 1, pp. 1–45, 2017.

[9] 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.

[10] 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.

[11] 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.

[12] M. Presa-Reyes, Y. Tao, S.-C. Chen, and M.-L. Shyu, "Florida international university-university of miami TRECVID 2020 DSDI track," in *TRECVID*. NIST, USA, 2020.

[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] S. Pouyanfar, Y. Yang, S. Chen, M. Shyu, and S. S. Iyengar, "Multimedia big data analytics: A survey," *ACM Computing Surveys*, vol. 51, no. 1, pp. 10:1–10:34, 2018.

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

[17] S. Pouyanfar, Y. Tao, A. Mohan, H. Tian, A. S. Kaseb, K. Gauen, R. Dailey, S. Aghajanzadeh, Y.-H. Lu, S.-C. Chen *et al.*, "Dynamic sampling in convolutional neural networks for imbalanced data classification," in *2018 IEEE conference on multimedia information processing and retrieval (MIPR)*. IEEE, 2018, pp. 112–117.

[18] S.-C. Chen, M.-L. Shyu, W. Liao, and C. Zhang, "Scene change detection by audio and video clues," in *Proceedings. IEEE International Conference on Multimedia and Expo*, vol. 2. IEEE, 2002, pp. 365–368.

[19] S.-C. Chen, M.-L. Shyu, C. Zhang, and R. L. Kashyap, "Identifying overlapped objects for video indexing and modeling in multimedia database systems," *International Journal on Artificial Intelligence Tools*, vol. 10, no. 4, pp. 715–734, 2001.

[20] 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.

[21] 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.

[22] 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.

[23] 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.

[24] 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.

[25] 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.

[26] 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.

[27] S. Pouyanfar and S.-C. Chen, "Automatic video event detection for imbalance data using enhanced ensemble deep learning," *International Journal of Semantic Computing*, vol. 11, no. 01, pp. 85–109, 2017.

[28] 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.

[29] S.-C. Chen, M.-L. Shyu, M. Chen, and C. Zhang, "A decision tree-based multimodal data mining framework for soccer goal detection," in *2004 IEEE International Conference on Multimedia and Expo*, vol. 1. IEEE, 2004, pp. 265–268.

[30] N. Rishe, J. Yuan, R. Athauda, S.-C. Chen, X. Lu, X. Ma, A. Vaschillo, A. Shaposhnikov, and D. Vasilevsky, "Semanticaccess: Semantic interface for querying databases," in *International Conference on Very Large Data Bases*, September 2000, pp. 591–594.

[31] S.-C. Chen, M.-L. Shyu, and R. L. Kashyap, "Augmented transition network as a semantic model for video data," *International Journal of Networking and Information Systems*, vol. 3, no. 1, pp. 9–25, 2000.

[32] 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.

[33] M. Presa-Reyes, Y. Tao, S.-C. Chen, and M.-L. Shyu, "Deep Learning with Weak Supervision for Disaster Scene Description in Low-Altitude Imagery," *IEEE Transactions on Geoscience and Remote Sensing*, accepted for publication, 2021.

[34] 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.

[35] 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.

[36] 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.

[37] M. Presa-Reyes, Y. Tao, R. Ma, S.-C. Chen, and M.-L. Shyu, "Multi-source weak supervision fusion for disaster scene recognition in videos," in *2022 IEEE 5th International Conference on Multimedia Information Processing and Retrieval (MIPR)*. IEEE, 2022, pp. 287–292.

[38] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *the 31th AAAI Conference on Artificial Intelligence*, 2017, pp. 4278–4284.

[39] 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.

[40] 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.

[41] 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.

[42] 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.

[43] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

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

[45] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*. PMLR, 2019, pp. 6105–6114.

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

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

[48] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Céspedes, S. Yuan, C. Tar *et al.*, "Universal sentence encoder," *arXiv preprint arXiv:1803.11175*, 2018.

[49] Q. Zhu, L. Lin, M.-L. Shyu, and S.-C. Chen, "Effective supervised discretization for classification based on correlation maximization," in *IEEE International Conference on Information Reuse and Integration*, 2011, pp. 390–395.

[50] K. V. Price, "Differential evolution," in *Handbook of optimization*. Springer, 2013, pp. 187–214.

[51] 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.