horizontal line

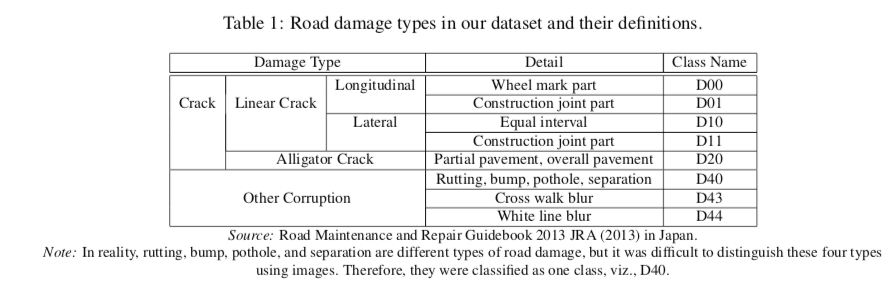
**Team: HAL - BDASL**

RDD 2020 IEEE Big Data Cup

**13th August 2020**

# OVERVIEW

Participants need to propose an algorithm that can automatically recognize the road damages present in an image captured from any of the following three countries: India, Japan, and Czech. The recognition implies the detection of the damage location in the image and the identification of the damage type.



* **RDD** Challenge: <https://rdd2020.sekilab.global/>
* **Data**set Link: <https://github.com/sekilab/RoadDamageDetector>

# GOALS

1. Create a basic toolbox using [**Detectron2**](https://detectron2.readthedocs.io/), [**MMDetection**](https://github.com/open-mmlab/mmdetection) and [**YoloV5**](https://github.com/ultralytics/yolov5)
2. Refer to **old dataset/model** for augmentation/init at <https://github.com/sekilab/RoadDamageDetector>
3. List out strategies for fine-tuning from old dataset to latest -> 2020
4. **Previous** challengers
   1. Open Images 2019 - Object Detection (https://github.com/Sense-X/TSD)
5. **Feature** Engineering
   1. Attend to the road bounds for all detection purposes
   2. **Class Balance** the dataset (RD2020 has a huge data imbalance problem)
   3. Ensemble models
   4. **Ensemble** boxed for detection <https://www.kaggle.com/c/open-images-2019-object-detection/discussion/115086>
   5. **GAN and VAE** for feature engineering
6. Categorically **train models** based on geography (czech, Japan and India). Produce inference on those three individually and submit the combined CSV
7. Run **AutoML** to identify the best Hyperparam/config
8. Use **VoVNet backend**. an upgrade from ResNet. <https://github.com/youngwanLEE/vovnet-detectron2>
9. Object Detection with **Transformers** https://github.com/facebookresearch/detr

# SPECIFICATIONS

Submission:Detection result upload to [https://rdd2020.sekilab.global/submissions](https://rdd2020.sekilab.global/submissions/)

Result and source code will also need to be submitted later from github

# MILESTONES

## Release Test1 – May 15th

Initial Template/Notebook Codebase developed on Google Colab to utilize free GPUs.

* [Detectron2](https://drive.google.com/file/d/1hILP40idiNIfQteBwBlXEAEQoECVUMo9/view?usp=sharing)
* [MMDetection](https://drive.google.com/file/d/1n33kxxWuEnr_WiMueTAkF5FL2MeEqsiI/view?usp=sharing)

## Setup D2 for Detection

* Prepare data split and submission result output with model versioning

## Setup Yolo or MMDet

## Release Test2 – Sept 10th (Sept 19th last submission date)