Document Approach EYChallenge

Our solution is based on the Phase-1 Benchmark Notebook provided by the EY Open Science Data Challenge Program.

First, we utilize the `rasterio` and `osgeo` libraries to access the TIF images of San Juan provided by the platform. These images are then divided into tiles for further processing and labeling of residences. Then we use the matlplot libraries to see the behavior of the model.

This process involves both pre-event and post-event images to identify residences that received damages post-event. Consequently, the generated images or tiles need to be renamed using the `imageCount` format. To achieve this, a different code is used compared to the one provided by the platform. Changing the tile names ensures that the order of the generated files remains consistent. This prevents difficulty in identifying pre-event and post-event images during labeling. A custom code is programmed to rename the tiles in such a way that the order remains intact. For example, if we search for `image1`, it corresponds to the same geographical position in both the pre_event and post_event folders. This streamlines the labeling process as we can easily identify if a residence was damaged after the event. Additionally, QGIS is employed to locate these residences and verify their bounding box to extract the most accurate label possible.

Next, the photos are annotated using LabelMe. We have four classes to label that are the following:

- **Damaged Commercial Building:** This class refers to buildings used exclusively for commercial purposes that exhibit structural damage. To identify such buildings, we examine the space around them, looking for indicators like parking spaces or a moderate presence of people, which could suggest commercial use. Damage is assessed primarily on the roof, where visible breaks or the presence of blue mesh/shade cloth indicate damage.
- **Damaged Residential Building:** This class applies to buildings used solely for domestic purposes that show structural damage. Identification involves examining the surrounding area for characteristics - - Undamaged Commercial Building: This class refers to commercially used buildings without visible structural damage. Identification criteria include the presence of parking or some level of people activity around the building, suggesting commercial use, without any apparent damage.
- **Undamaged Residential Building:** This class denotes buildings used for domestic purposes without visible structural damage. To identify these buildings, we look for features like significant land or adjacent houses with similar structures, indicative of residential use, and the absence of apparent damage.

As mentioned previously, we utilize QGIS software because it allows us to open .tiff images. We then import the Pre-Event and Post-Event images as well as the building footprint as layers. These image layers enable us to observe the changes in buildings pre- and post-disaster at the same location, while the footprint aids in visualizing the building's shape, making it easier to label within the boxes. This step is one of the most important, as it was crucial in verifying the accuracy of our labels.

With image processing complete, `labelme2yolo` is utilized to convert the annotations made with LabelMe into a format recognizable by YOLO.

The model chosen for this task is `yolov8m-obb`, primarily because it demonstrated superior performance compared to other YOLO models tested. However, we also explored the implementation of the Mask R-CNN model, renowned for its effectiveness in satellite image computer vision tasks. Extensive research indicated that Mask R-CNN could potentially provide even better results than YOLO due to its architecture and the utilization of ResNet101 as its backbone.

Unfortunately, due to time constraints and compatibility issues with TensorFlow versions, the implementation of Mask R-CNN remained incomplete. Despite making progress in its implementation, we were unable to fully integrate it into our workflow within the available timeframe. It's worth noting that the Mask R-CNN model excels in tasks requiring precise object detection and segmentation, attributes highly relevant to our project's objectives.

Returning to the YOLO model, after selecting `yolov8m-obb`, we further optimized its performance by adjusting parameters and hyperparameters. This optimization included fine-tuning parameters such as learning rate and optimizer selection. Through experimentation, we observed that the AdamW optimizer yielded the best results for our specific task. The optimization process aimed to enhance the model's ability to accurately detect and classify objects, particularly residences, in both pre-event and post-event imagery of San Juan.

While experimenting with hyperparameters, parameters aimed at increasing model diversity were modified. These included `mixup`, `erasing`, `translate`, `scale`, `flipud`, `fliplr`, `mosaic`, and `degrees`. These parameters enhance the model's generalization ability and improve its performance in recognizing objects in various orientations and contexts. The model was trained for 85 epochs, which yielded the best results when validating the data with the submission images.

The parameters were fine-tuned based on visualizations stored in the `runs` folder generated after each model training. This folder also contains a .yaml file with the model configurations at the time of training. We obtained the following results:

Category	mAP50
all	0.531
damagedcommercialbuilding	0.492
damagedresidentialbuilding	0.619
undamagedcommercialbuilding	0.235
undamagedresidentialbuilding	0.777

After training the model, submission images downloaded from the platform are used to validate the data and upload them to the platform to verify the model's accuracy.

For more information, you can find the source code along with the documentation on the following github: https://github.com/natalia-romero/ey-challenge