**Report on Wind machine inspection Assignment**

**Introduction:**

This report details the creation and training of a YOLO (You Only Look Once) object detection model utilizing a custom dataset. The dataset comprises 47 preprocessed images categorized into three classes: corrosion, scratch, and normal. Due to resource limitations of my side, the dataset size was constrained to 47 images from an initial pool of 700. The decision to limit the dataset size was influenced by challenges encountered in the Google Colab environment, including high training times and difficulties in uploading large files.

**Challenges Faced:**

The primary challenge encountered during the project was the collapse of Google Colab instances due to extended training times. Training deep learning models, particularly YOLO, demands significant computational resources, and the limitations of available GPU resources in Google Colab became apparent. The extensive training times required for YOLO, exacerbated by the model's complexity and the dataset size, led to instability in the Colab environment.

Furthermore, the slow internet connectivity hindered the efficient uploading of large datasets. The initial dataset of 700 images posed challenges in terms of data transfer speeds, making it impractical to upload the entire dataset to the Google Colab environment for training. As a result, a smaller subset of 47 images was chosen for the training process to mitigate these challenges.

**Dataset Annotation:**

To annotate the dataset, the Computer Vision Annotation Tool (CVAT) was employed. This tool facilitated the annotation of objects in images, defining bounding boxes for each instance of corrosion, scratch, or normal class. The annotations were exported into a JSON format compatible with YOLO requirements.

**YOLO Model Overview:**

YOLO is a real-time object detection system that can detect multiple objects in an image in a single forward pass. It divides the image into a grid and predicts bounding boxes and class probabilities for each grid cell. YOLO is known for its speed and efficiency, making it suitable for real-time applications.

The YOLO model architecture comprises several convolutional layers, ultimately predicting bounding boxes, confidence scores, and class probabilities. The model divides the input image into a grid, and for each grid cell, multiple bounding boxes are predicted. Non-maximum suppression is then applied to filter out redundant predictions.

**Training the YOLO Model:**

Using the annotated dataset, the YOLO model was trained. The annotated images and their corresponding bounding box information were used to optimize the model for detecting corrosion, scratches, and normal instances. Training involved adjusting model weights based on the difference between predicted and ground truth bounding box coordinates and class probabilities.

**Results:**

Despite the limited dataset, the trained YOLO model demonstrated promising results. During testing, it correctly predicted the presence of corrosion in two out of the three corrupted images. The model's ability to generalize to new, unseen data is crucial for its practical utility.

**Conclusion:**

In conclusion, the custom YOLO object detection model, trained on a small but carefully annotated dataset, shows potential for detecting corrosion images. The limitations of a small dataset and computing resources should be considered, and further refinement and evaluation on a larger dataset would likely improve model performance.