This project introduces the Yolo8 Annotation Tool, a user-friendly Python application designed to streamline the creation of annotated datasets for object detection tasks. Developed with the PyQt6 library, this tool empowers researchers and developers to efficiently label images with bounding boxes, a crucial step in training YOLOv8 and other machine learning models. The tool incorporates features such as bounding box drawing, undo/redo, and annotation management.

**Goals:**

* To provide a robust and easy-to-use annotation tool that enhances the efficiency of dataset preparation for object detection tasks.
* To support users in producing high-quality annotated datasets that improve the performance of machine learning models.

**Key Features:**

* **User-Friendly Interface:** The application offers an intuitive GUI built with PyQt6, making it accessible for users of all skill levels.
* **Bounding Box Annotation:** Users can easily draw, modify, and delete bounding boxes around objects within images, facilitating precise annotations.
* **Batch Processing:** Users can load multiple images for annotation, speeding up the dataset creation process.
* **Image setting:** Users can easily resize, vary brightness, contrast, and rotate to introduce data argumentation.
* **Undo/Redo Functionality:** Mistakes can be quickly corrected with undo and redo features, ensuring a smooth workflow.
* **PNG Converter:** Image with any format can be converted to PNG format for easy handling of image format.
* **Dataset Splitter:** After annotation split the images into training, validation and testing dataset.

**Target Users:**

* This tool is aimed at data scientists, researchers, and machine learning practitioners who require a reliable method for creating annotated datasets, particularly for projects focused on object detection.

**Technological Stack:**

* Programming Language: Python
* GUI Framework: PyQt6
* Image Processing: Libraries such as OpenCV or PIL for handling image operations.

**Limitation**

* **Single Category Annotation:** This tool currently supports annotating images with only one category or class per project.
* **Single Format Annotation:** This tool currently supports annotation in a single format compatible with the YOLOv8 model.

**Annotation format**

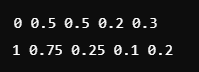
The YOLOv8 model annotation format typically consists of text files with the same name as the corresponding image file. Each line in the text file represents one object in the image and follows this structure:

[](https://github.com/RajeshRamadas/Yolo8-Annotation-Tool/blob/main/image/doc/img.png)

Explanation of the Components:

* **class\_id:** An integer representing the category of the object.
* **x\_center:** The x-coordinate of the bounding box center, normalized to the image width (value between 0 and 1).
* **y\_center:** The y-coordinate of the bounding box center, normalized to the image height (value between 0 and 1).
* **width:** The width of the bounding box, normalized to the image width (value between 0 and 1).
* **height:** The height of the bounding box, normalized to the image height (value between 0 and 1).

Example: For an image named image1.jpg, the corresponding annotation file would be image1.txt and might contain:



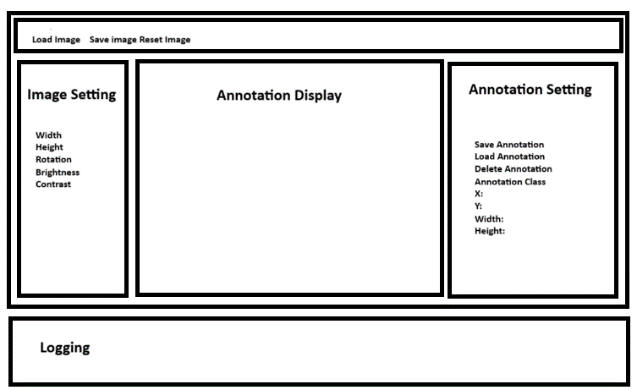
**In this example:**

The first line represents an object of class 0 with a bounding box centered at (50%, 50%) of the image, with a width of 20% and height of 30%.

The second line represents an object of class 1 with a bounding box centered at (75%, 25%).

This format is essential for training YOLOv8 models effectively.

**Yolo8 Annotation Tool GUI:**



**Conclusion**

The Yolo8 Annotation Tool is designed to be user-friendly and efficient, providing a robust set of features for creating annotated datasets for machine learning training. It simplifies the process of image annotation, making it accessible for both novice and experienced users in the field of computer vision.