Object detection model to recognize cat and dog in an image

Here is a brief description of my experiment and some observations from it

Description:

- **Dataset preparation**: Prepared a custom dataset of 100 images containing dogs and cats. This involved manually labeling the images with bounding boxes and class labels in the YOLO format using the GitHub software **labelImg-master**. then split the dataset into training, validation, and testing sets with 70%, 15%, and 15% of the images, respectively.

- **Model training**: Trained the **YOLOv8.0.20** model using the **Ultralytics** framework on my custom dataset. This involved setting up the configuration files, specifying the training parameters such as the number of epochs, and training the model on the training set.

- **Model validation**: Validated the performance of the trained model on the validation set during the training process. This helped monitor the training progress and adjust the hyperparameters if necessary.

- **Model testing**: Evaluated the performance of the trained model on the testing set after the training process was completed. This involved calculating the precision, recall, and average precision, as well as visualizing the output of the model on some sample test images to check the accuracy of the detections.

Observations:

- **Dataset size**: With only 100 images, the size of the dataset is relatively small. This may limit the performance of the model, especially if the images are highly variable in terms of lighting conditions, background, and pose.

- **Hyperparameters:** The performance of the model may depend on the hyperparameters used for training, such as the learning rate, batch size, and the number of epochs. Therefore, it is important to experiment with different configurations and settings to achieve the best results.

- **Class imbalance**: If the number of images with cats and dogs is not balanced in the dataset, the model may be biased towards one class or the other. Therefore, it is important to ensure that the dataset has a roughly equal number of images for each class.

- **Object size:** The accuracy of the model may be affected by the size of the objects in the images. If the objects are too small or too large, the model may not be able to detect them accurately. Therefore, it is important to choose an appropriate input image size and anchor boxes for the YOLOv8.0.20 architecture.

Tools for the experiment:

- labelImg-master. [For image annotations with bbox]

- Google drive. [cloud storage]

- Google collaboratory.

Conclusion:

Overall, the experiment is a good starting point for building a dog and cat detection system using Ultralytics YOLOv8.0.20. By fine-tuning the model and experimenting with different configurations and settings, can improve the accuracy and robustness of the system.

---0---