**Object Detection with Faster R-CNN: Technical Documentation**

**Introduction**

Object detection is a fundamental task in computer vision, enabling machines to identify and localize objects within images. This document presents an in-depth exploration of building an object detection model using the Faster R-CNN algorithm and its application in real-world scenarios. The app will also utilize bounding boxes around detected objects and add labels for it.

In today's digital era, the ability to automate object detection tasks holds immense value across various domains. This project addresses the need for accurate and efficient object detection methods, crucial in applications like surveillance, autonomous vehicles, and industrial automation.

**Selected Machine Learning Algorithm**

For detecting objects, the chosen algorithm will be Faster R-CNN (Region-based Convolutional Neural Networks) which is a modern deep learning model specifically designed for object detection tasks. Its ability to accurately contain and classify objects in an image makes it an ideal choice for this project.

**Development of the Model**

The model development involved several key steps. Firstly, before starting the development, the student created a virtual environment inside Visual Studio Code by inputting “python -m -venv [path of the project]\venv” and then installing various libraries needed for the app via pip. The libraries included are streamlit, for hosting it via web application (localhost), PIL for the interface, NumPy for mathematical purposes, matplotlib for graphs, and torch for importing Faster R-CNN and such.

The selection of a pre-trained Faster R-CNN model is important, leveraging its learned features from a vast dataset. Fine-tuning the model with a specific dataset tailored for our object detection needs ensured its adaptability to the task at hand. The model was trained using a dataset encompassing diverse object classes, ensuring robustness and versatility.

**Results and Performance**

The trained Faster R-CNN model demonstrated promising performance in object detection. With a box score threshold of 0.8, it accurately localized objects with an average precision of over 85% across various classes. Visualizations showcased the model's ability to detect and classify objects accurately, albeit with certain limitations in smaller or occluded objects like the model only accepts images with a resolution of 590x590, as inputting more than that gives the error. Pixel art can’t also be detected.

**Critical Reflection**

Reflecting on the project, while the Faster R-CNN model exhibited commendable performance, challenges were encountered in handling smaller objects and instances with heavy occlusions. Fine-tuning hyperparameters and exploring alternative architectures could potentially address these limitations.

**Conclusion**

In conclusion, the utilization of Faster R-CNN for object detection proved effective, highlighting its strengths in accurately localizing and classifying objects within images. However, further refinement and exploration of model enhancements remain vital for tackling challenging scenarios.