VILLAFLOR, JUSTIN JOHNN MICO

CCST106-4A

Computer Vision

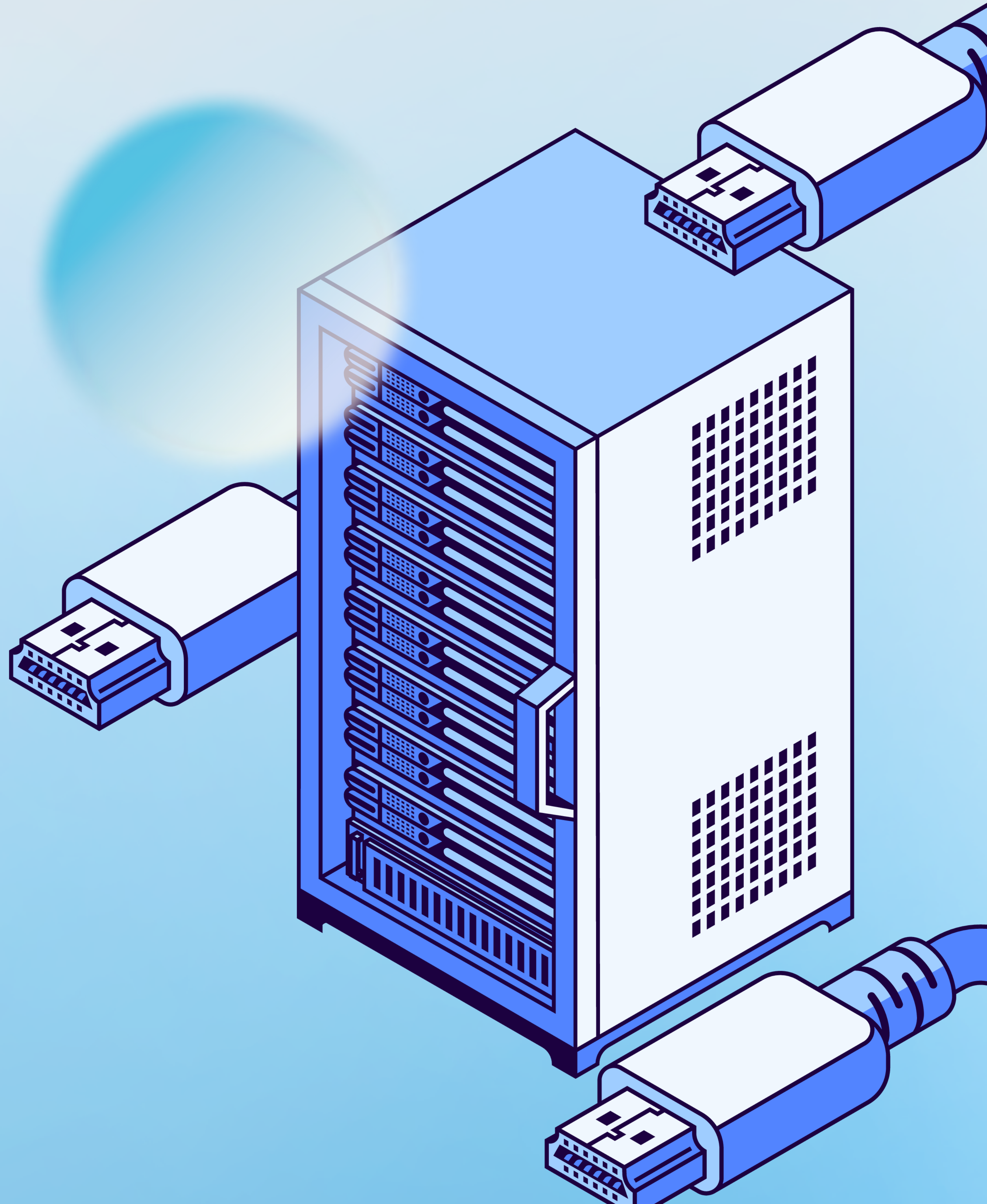
Introduction

In this project, I compare two popular object detection models: YOLOv8 and SSD (MobileNetV2). Object detection is a critical task in computer vision, where models are trained to recognize and locate objects in images or video. The aim of this report is to evaluate the performance of these models using a common dataset, identify their strengths and weaknesses, and provide insights into their practical applications.

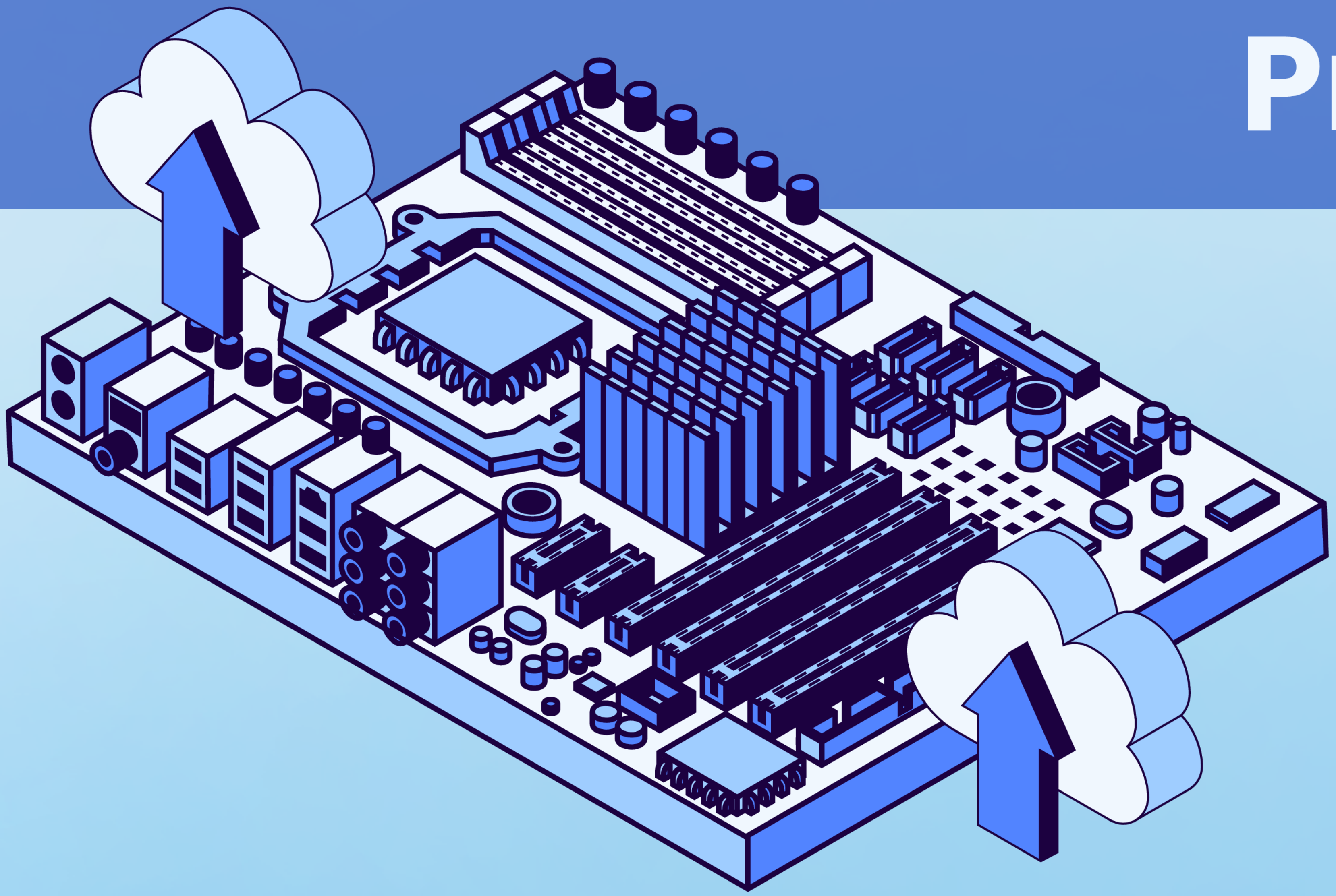
Both models are renowned for their performance in real-time object detection, with YOLOv8 being optimized for high-speed, real-time processing, and SSD (MobileNetV2) being designed for resource-efficient environments like mobile and embedded systems.

Dataset and Data Preparation

For this project, I used the COCO (Common Objects in Context) dataset, a widely-used benchmark for object detection tasks. COCO was selected because it contains diverse objects, including humans, animals, vehicles, and furniture, making it suitable for evaluating the generalization capabilities of both models.



Preprocessing Steps



Resizing

For YOLOv8, all input images were resized to 640x640, while SSD required images to be resized to 300x300 (or 512x512) to match the input dimensions required by the respective models.

Normalization

Each pixel's value was scaled between 0 and 1 to ensure faster convergence during training.

Bounding Boxes

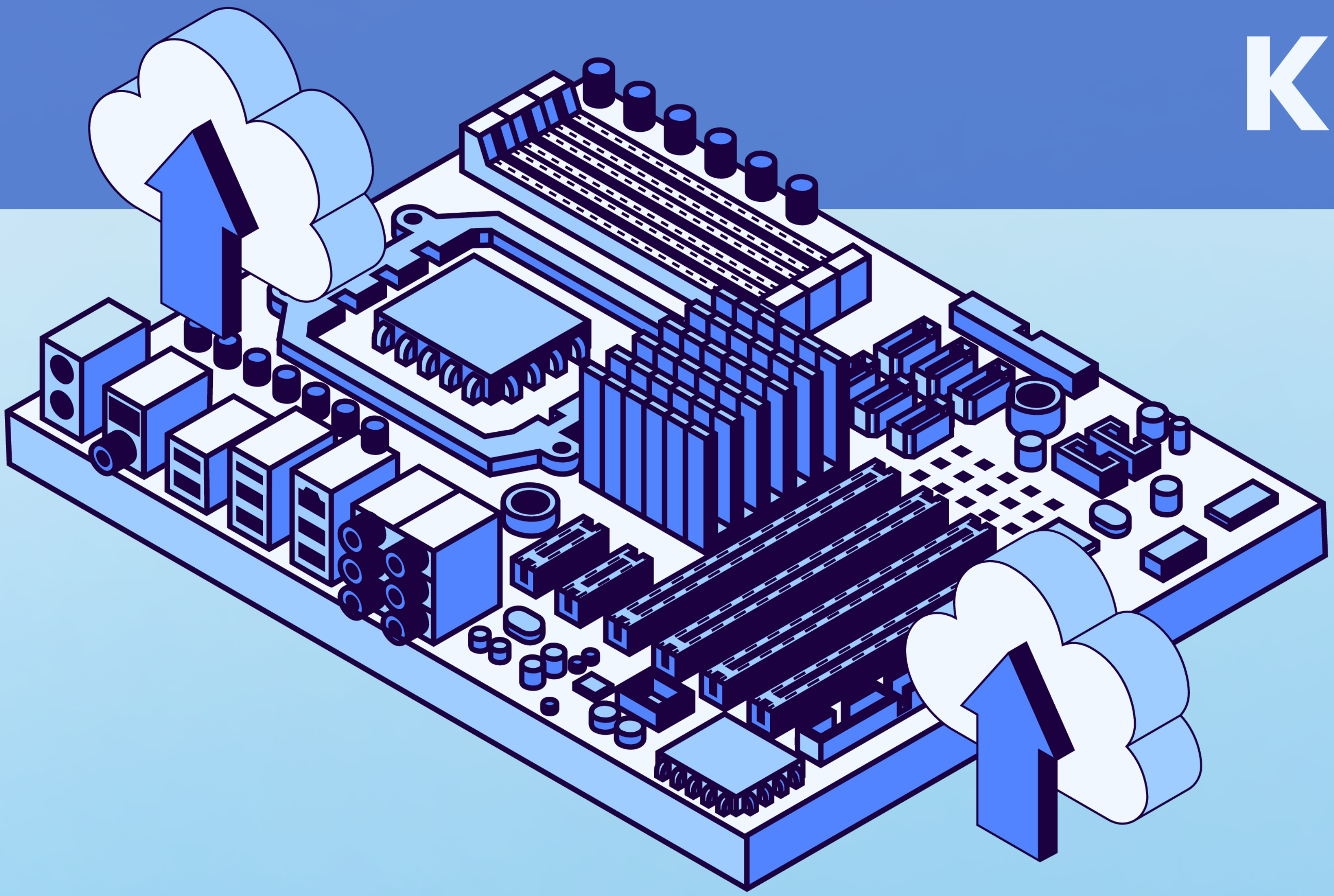
For both models, annotations were processed to apply bounding boxes and labels. These bounding boxes define the regions of interest that contain objects in the images.

Model Implementation

YOLOv8 Model

YOLOv8 is the latest iteration in the YOLO family of object detection models. It is designed for fast and accurate detection by using a single neural network to predict both bounding boxes and object class probabilities in one forward pass. The YOLOv8 model architecture includes a CSPDarknet backbone that enhances feature extraction and improves detection accuracy.
We implemented YOLOv8 using the PyTorch framework, leveraging a pre-trained model and fine-tuning it on the COCO dataset.

Key Features of YOLOv8



Single-pass detection

This approach allows the model to predict bounding boxes and class probabilities in one forward pass through the network, significantly speeding up the detection process.

Optimized for real-time applications

This means the model is designed to deliver fast inference speeds, enabling it to perform object detection in real time without noticeable delays.

Supports dynamic anchor boxes for better accuracy

This feature allows the model to automatically adjust the sizes and ratios of anchor boxes during training, improving the accuracy of object localization and classification.

Model Implementation

SSD (MobileNetV2) Model

combined with the MobileNetV2 backbone, is an efficient model for object detection on resource-constrained devices. SSD is designed to detect objects of various sizes using multiple convolutional layers, each detecting objects at different scales. MobileNetV2 acts as the backbone, providing a balance between speed and performance, making it ideal for mobile applications.

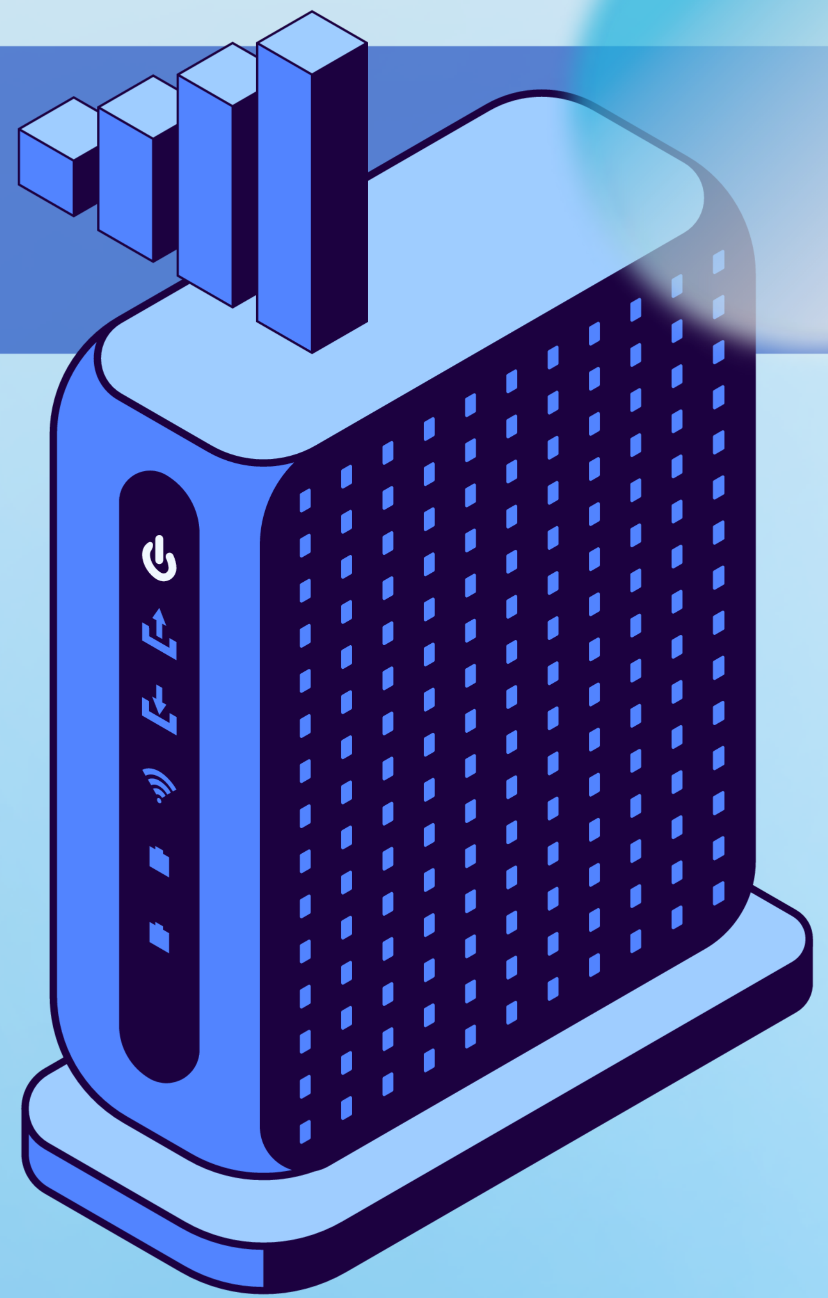
Key Features of SSD (MobileNetV2)

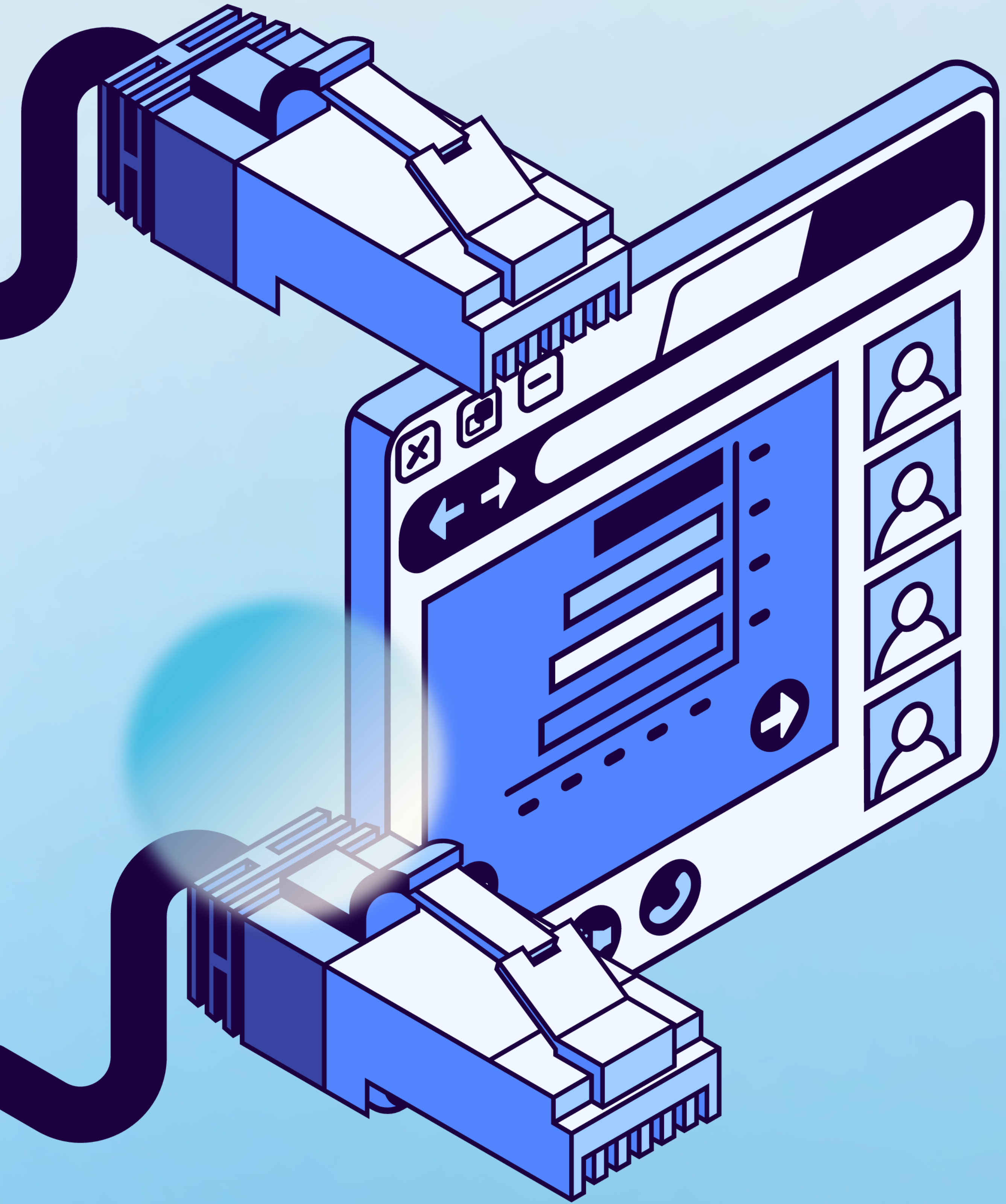
Multi-scale detection

This technique enables the model to detect objects of varying sizes by using feature maps at multiple resolutions, allowing it to identify both small and large objects effectively.

Efficient for mobile and embedded systems

This indicates that the model is designed to run with low computational power and memory usage, making it suitable for deployment on devices like smartphones and IoT devices.





Performance Metrics

I evaluated both models using standard performance metrics, including Precision, mAP (Mean Average Precision), Speed, and Latency.

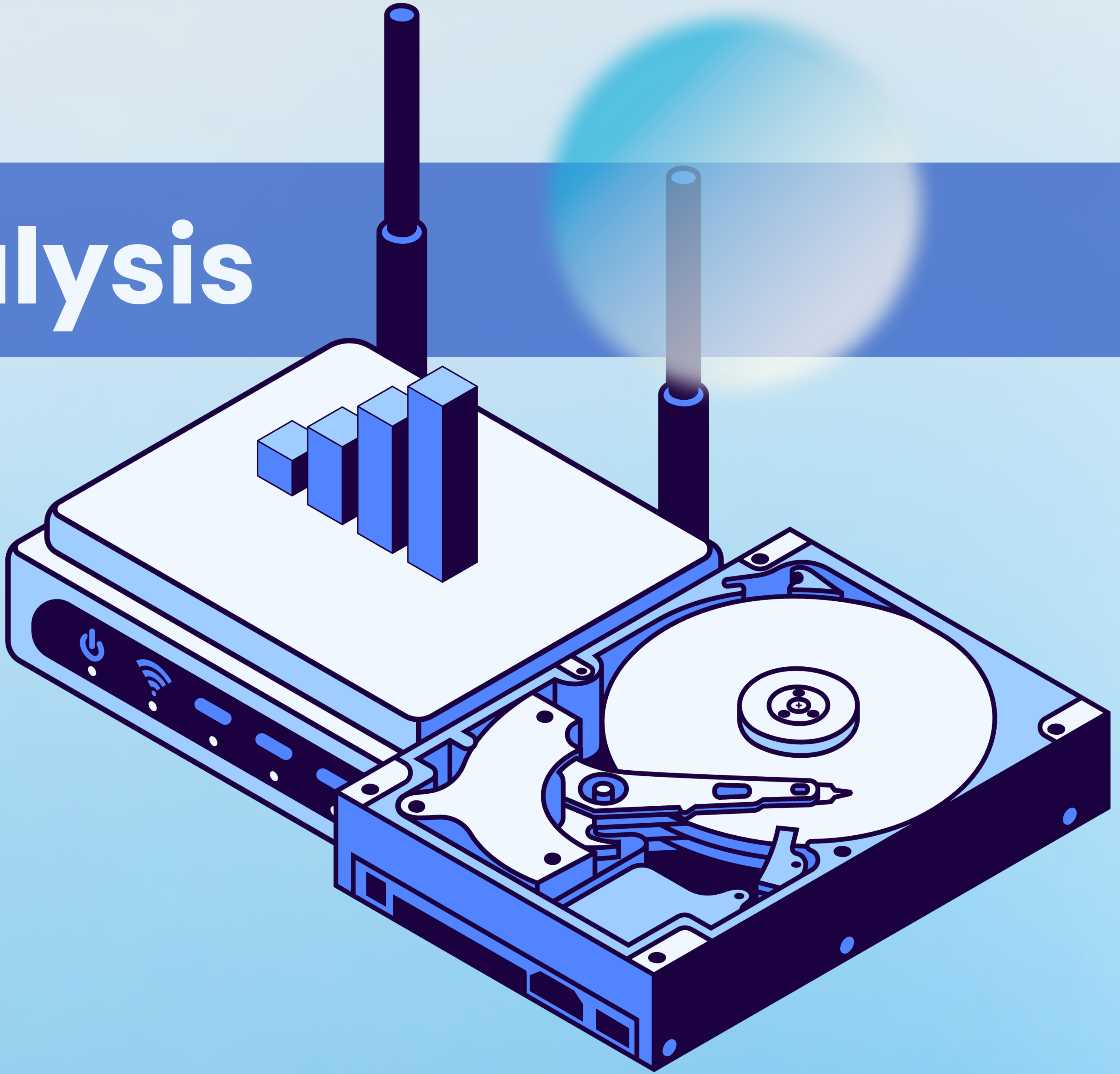
YOLOv8 Performance

- Precision: 0.6 to 0.9 (using Chess Pieces dataset)
- mAP: ~50% on COCO dataset
- Speed: 50-140 FPS
- Latency: ~10 ms on cpu

SSD (MobileNetV2) Performance

- Precision: 0.5 to 0.7
- mAP: ~22-25% on COCO format Chess Pieces dataset
- Speed: 20-30 FPS on mobile devices

Comparative Analysis



Speed and Efficiency

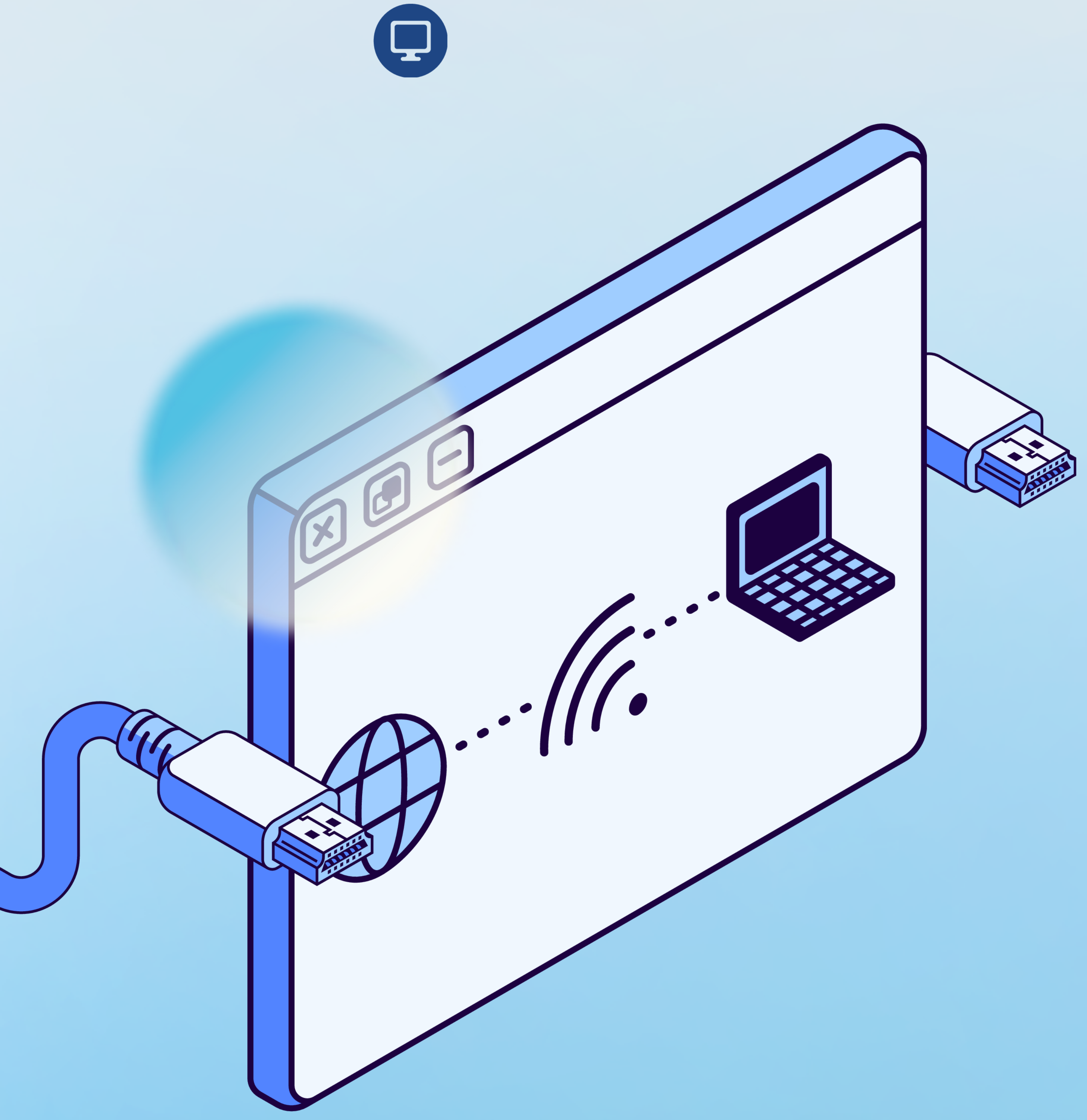
- YOLOv8: Faster than SSD, real-time performance with single-pass detection.
- SSD (MobileNetV2): More efficient in low-resource environments but slower on larger images and complex tasks.

Detection Accuracy

- YOLOv8: Higher accuracy, better at large object detection, flexible input sizes.
- SSD (MobileNetV2): Better at small object detection due to multi-scale anchor boxes but lower overall accuracy.

Resource Utilization

- YOLOv8: Requires more computational resources but scalable to edge devices with lighter models.
- SSD (MobileNetV2): Optimized for mobile platforms, lower resource consumption.



Discussion of Challenges

During this project, we faced challenges primarily related to data preprocessing and model tuning. The COCO dataset presented difficulties with varying image resolutions and formats, so we implemented a robust preprocessing pipeline that included consistent resizing and normalization to prepare the data for training.

For model tuning, particularly with YOLOv8, the extensive hyperparameter options made it challenging to find optimal configurations. We addressed this by conducting experiments with various learning rates, batch sizes, and anchor box settings, using cross-validation to enhance both accuracy and speed.

Through these challenges, we learned the critical importance of careful data preparation and iterative experimentation in achieving effective object detection, deepening our understanding of how each component influences model performance.

VILLAFLOR, JUSTIN JOHNN MICO

Conclusion and Next Steps

In summary, this project involved selecting the COCO dataset, implementing two object detection models—YOLOv8 and SSD (MobileNetV2)—and evaluating their performance based on speed and accuracy. We thoroughly prepared the dataset through consistent preprocessing, followed by training both models and analyzing their detection capabilities.

VILLAFLOR, JUSTIN JOHNN MICO

Thank You!

For Listening