Good morning, everyone. We are the UAV Object Detection Group. My name is Hu Kaiwei, and I am here to deliver a presentation on behalf of my team.

Today, we will go through what we have learned and accomplished in our project. Our presentation will be divided into three parts:

1. The YOLO command structure
2. The NSCC PBS script
3. Our testing results

Here, we chose the VisDrone dataset, specifically Task 1: Object Detection in Images. We utilized the YOLO11 framework to train and test our models.

YOLO Command Structure:

For the first part, we'll discuss about the YOLO command pattern. We used the YOLO command to train, validate pre-trained YOLO models and test, benchmark the fine-tuned versions.

The YOLO command follows this specific pattern:

yolo TASK MODE ARGS

- **TASK** (optional) can be one of: 'segment', 'obb', 'detect', 'classify', or 'pose.'
- **MODE** (required) can be: 'predict', 'track', 'benchmark', 'val', 'export', or 'train.'
- **ARGS** (optional) are custom parameters like 'imgsz=320' that override defaults.

TASK and MODE are the options, and ofc arguments are the arguments.

Different tasks and modes require different arguments, including hyperparameters for training.

NSCC PBS Script:

For the second part, due to the limitation of GPU resources, we utilized GPU resources from the National Supercomputing Center. To do this, we learned how to use PBS scripts. We

learned basic PBS commands and how to submit job script files to the scheduler. Here are some images showing this process.

Testing Results:

The third part is our testing results. Here, We use five yolo11 object detection models shown below.

Performance Metrics:

Before diving into the results, let’s introduce some key performance concepts. What can be considered good performance?

- **Precision:** This measures the accuracy of positive predictions. Higher values are better.
- **Recall (or Sensitivity):** Indicates how well the model identifies all relevant objects. Higher values are better.
- **mAP50 (mean Average Precision at IoU=0.5):** Higher values indicate more accurate detections. Typically, an mAP50 score of 50% or above is considered good, depending on the task's complexity.
- **mAP50-95:** This is the average mAP across IoU from 50% to 95% with a step size of 5%. It’s a stricter metric, and values above 30% are considered good, especially for complex datasets.

"Now, let's move on to our testing results. We evaluated the performance of five fine-tuned YOLO models: YOLOv11n, YOLOv11s, YOLOv11m, YOLOv11l, and YOLOv11x."

"For training, we used the following command pattern:"

yolo detect train model=xxx.pt data=VisDrone.yaml batch=64 epochs=100 device=0,1,2,3 imgsz=640

"Similarly, for testing, we used this command:"

yolo detect val model=/home/users/ntu/khu005/scratch/project/yolo11*/runs/detect/train/weights/best.pt data=VisDrone-test.yaml batch=64 epochs=100 device=0,1,2,3 imgsz=640

"And to 'analyze the effect of training duration, we increased the epochs for YOLOv11x from 100 to 200. This allowed us to compare how extended training impacts both the training and testing results."

Findings:

Here, we mainly focus on the last two columns of each table and inference time.

We concluded that as the model complexity increased from nano to extra-large, performance improved, but inference time also increased.

Next, we compared the YOLOv11x testing results between 100 epochs and 200 epochs. We observed that the model's overall performance slightly decreased when training increased from 100 to 200 epochs, which means the possible overfitting after 100 epochs.

We also compared the YOLO11x training results between 100 and 200 epochs. While longer training improved performance on the training dataset, it led to overfitting, which reduced performance on unseen data.

Conclusion:

In summary, while longer training improves performance on the training dataset, it can lead to overfitting, which reduces performance on unseen data. Finding the right balance in training epochs is essential for optimal model performance.

That’s all for our presentation. Thank you for listening!