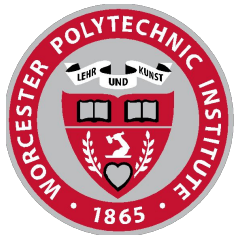



WPI

# Object Detection and Depth Estimation for Drone Camera Images

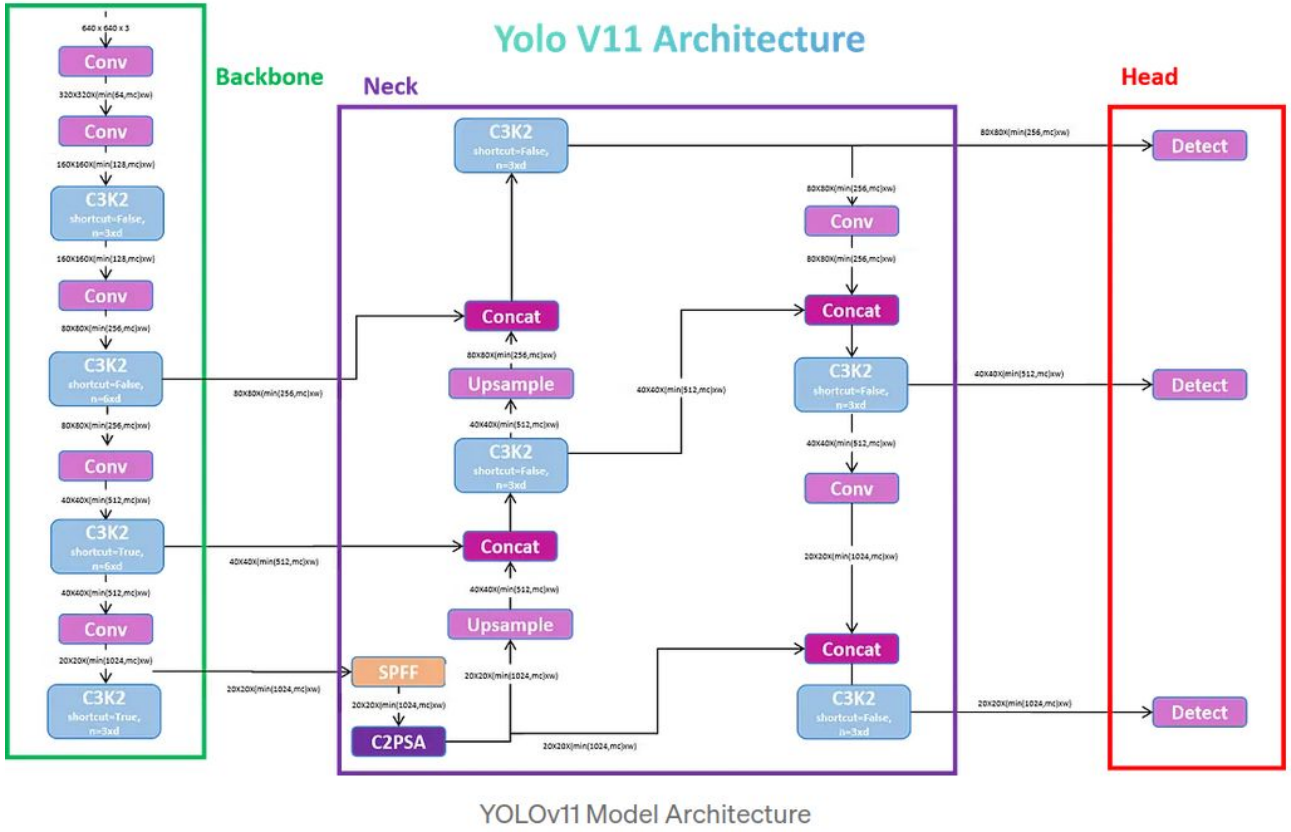
Paul Crann, Eric Viscione

# Object Detection Using Yolo V11

You only look once

# YoloV11 - Model

- Real Time Detection Model
- A single neural network is applied to a whole image
  - Single Neural network = You Only Look Once
  - This network will internally subdivide the image then predict the bounding boxes per region
- Many Pretrained models are provided which speeds up the training purposes
  - We used the yolo11m pretrained model



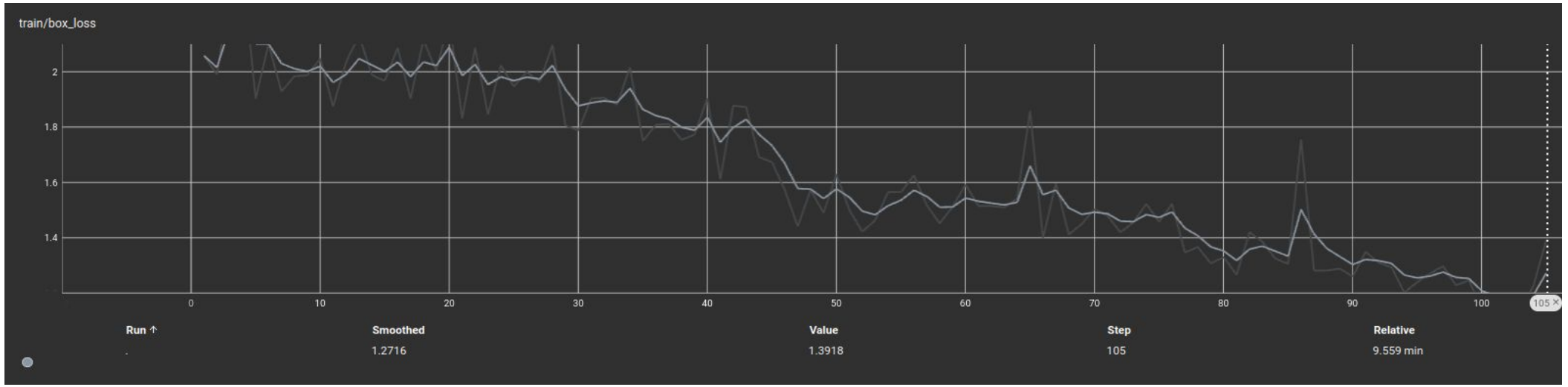

YOLOv11 Model Architecture

# Training Data

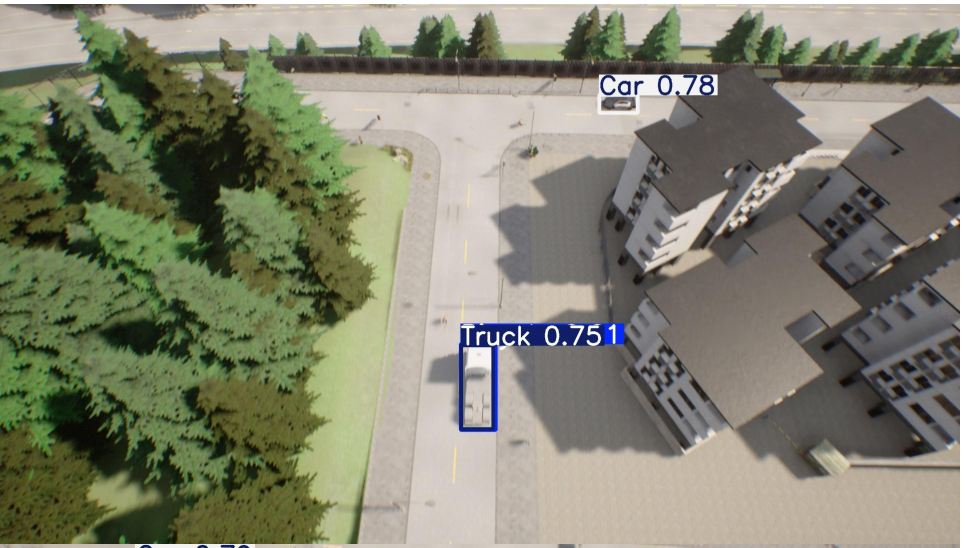
- Images from the Syndrone dataset were used
  - Town01 RGB data at heights of 20, 50 and 80 meters
- Each image needed to be annotated
  - Used the label-studio applet
  - Categories used: Car, Bus, Motorcycle, Bicycle, Truck

# Training

- Dataset was loaded and pointed the trainer to the images and the bounding boxes
  - The training set was 100 labeled images with 400-500 labeled objects



# Results


Car 0.78
Truck 0.751


Car 0.79
Truck 0.73
Truck 0.82
Truck 0.60

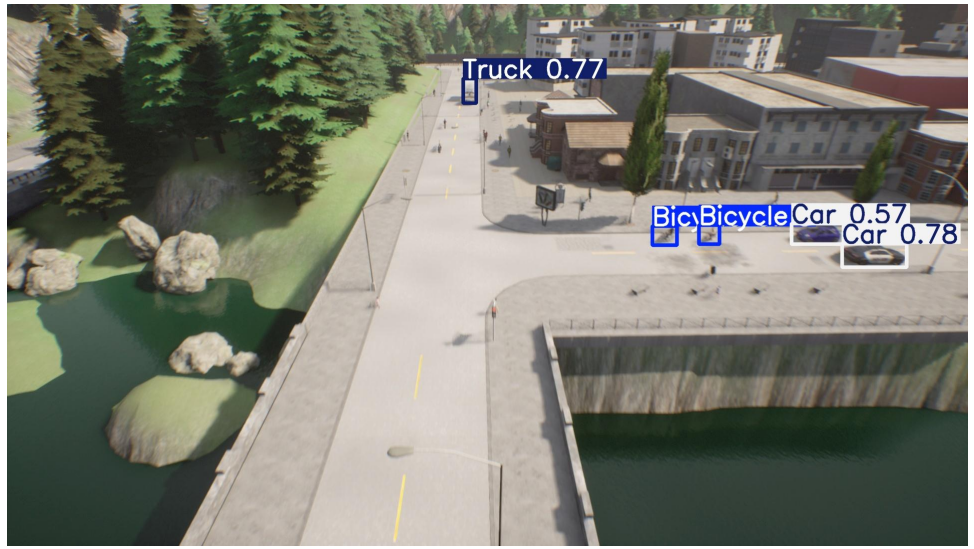

Car 0.27
Car 0.36
Motorcycle 0.62

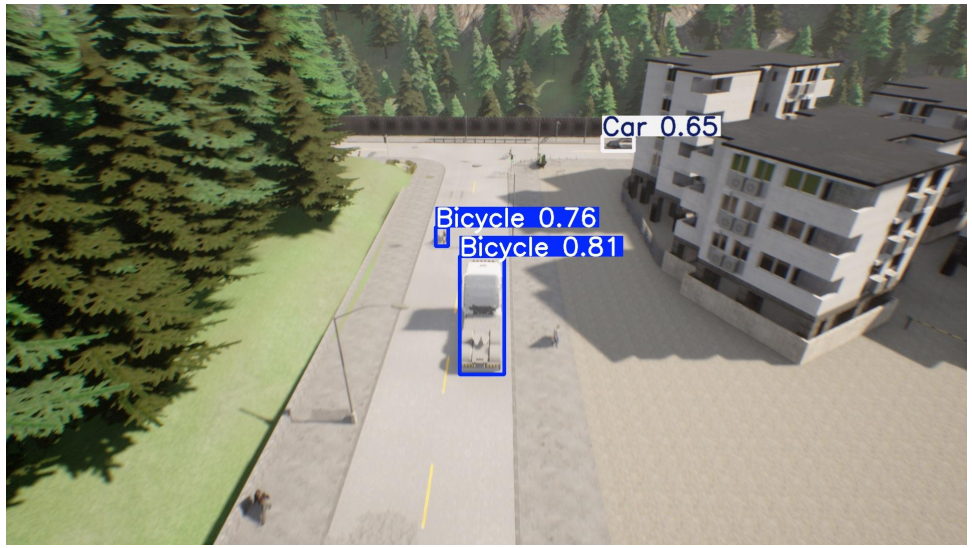
# Results


Bus 0.84
Car 0.65
Car 0.77
Motorcycle 0.41
Car 0.69


Car 0.77
Car 0.79
Motorcycle 0.77
Truck 0.67
Car 0.65
Car 0.64

# Results and Discussion



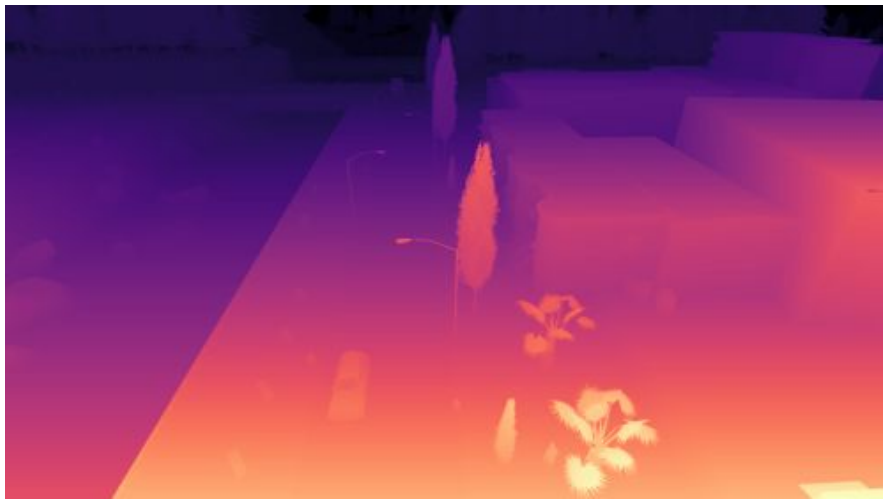

- The model was able to generally accurately identify the types of vehicles in each scene
  - The inaccuracies could be mitigated with an increase in the training data size

# Dense Prediction Transformer for Syndrone Dataset

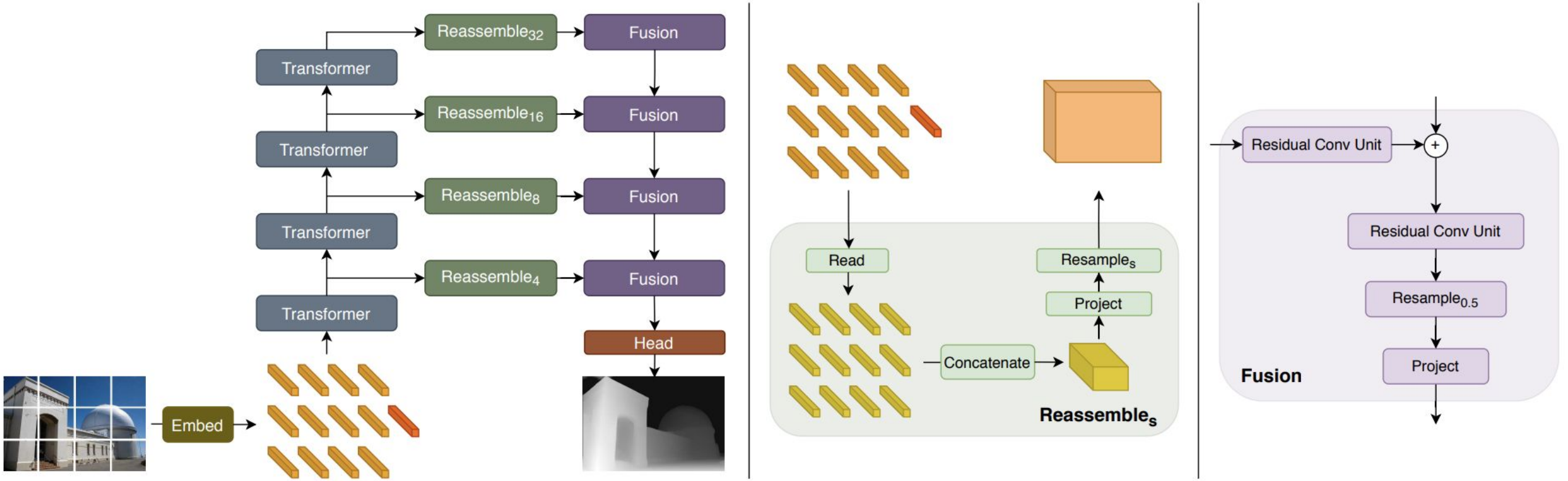
Monocular Depth Prediction

# Overview

- Fine tuned monocular depth estimation models proposed in R. Ranftl, et. al, “Vision Transformers for Dense Prediction”, 2021
  - https://arxiv.org/pdf/2103.13413
- Supervised approach on dataset provided by G. Rizzoli, et. al, “SynDrone - Multi-modal UAV Dataset for Urban Scenarios”, 2023
  - https://github.com/LTTM/Syndrone/tree/main



# Model Overview - DPT Large



# Training - DPT Large

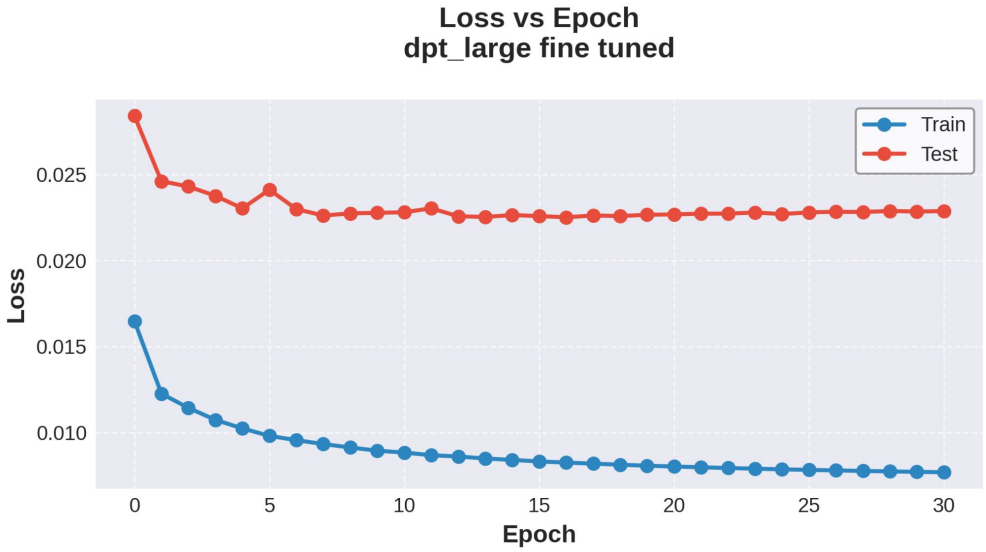
- Eigen Loss function proposed in Eigen et al. “Depth Map Prediction from a Single Image using a Multi-Scale Deep Network” 2014.
  - Computes the scale invariant MSE loss
  - https://arxiv.org/pdf/1406.2283
- Robust regression (Huber loss) was used to calibrate pretrained model outputs to our dataset (m=2.52e-5 b=3.33e-5).

```python
def eigen_loss(outputs, truths, lam=0.5):
    # Input: output and truth values (1/distance)
    outputs_d = 1 / (outputs + 10**-4.5)
    truths_d = 1 / (truths + 10**-4.5)

    # Scale-Invariant MSE + L2
    d = torch.log(outputs_d) - torch.log(truths_d)
    n = d.numel()
    scale_invariant_MSE = torch.sum(d**2) / n - lam*(torch.sum(d)/n)**2

    return scale_invariant_MSE
```

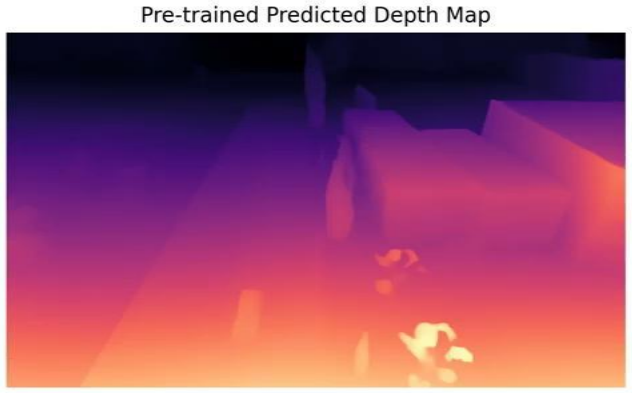
| Hyper Parameter | Value |
|---|---|
| **Initial Learning Rate** | 1e-5 |
| **LR Gamma** (Exponential Decay) | 0.93 |
| **Batch Size** (Memory Constraint) | 1 |
| **Epochs** | 30 |

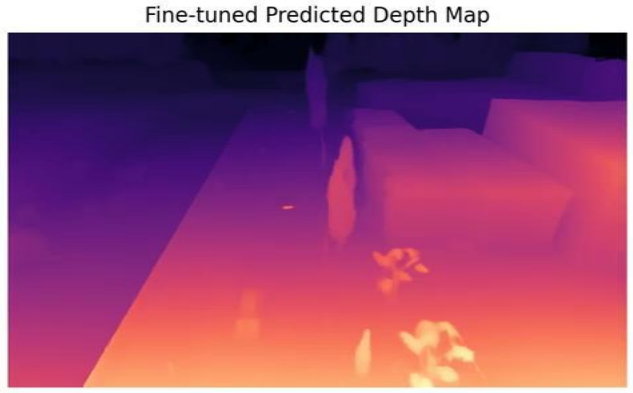
# Results - DPT Large

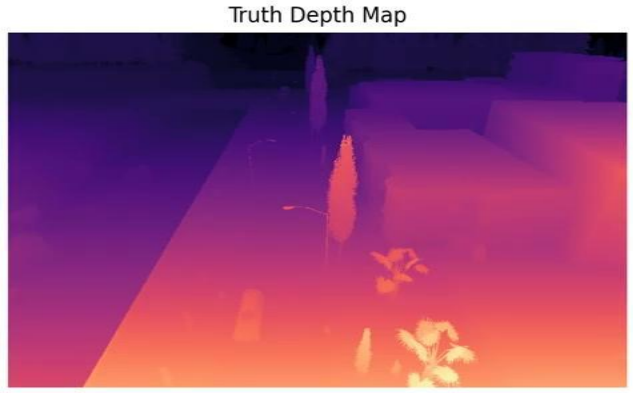
- Achieved an improved depth prediction model on unseen testing data
- Optimal model at epoch 16







# Results Cont - DPT Large