A Simple Demonstration of Camera Only Perception-Navigation with Metric3D for an Indoor Robot #4830

Owen-Liuyuxuan started this conversation in **Show and tell**

Owen-Liuyuxuan on Jun 6 Collaborator

Hello Community,

We are excited to share an update on our internal training project at TIER IV, where we are developing a camera-only perception navigation stack for indoor robots, utilizing Autoware and a monocular depth prediction model, specifically Metric3D. This model converts images from a single camera into dense point clouds, which then serve as the sole input for the Autoware Perception stack. Our aim is to assess the efficacy of the camera-only perception and planning modules within an indoor robotic setup.

Overview of Metric3D

Metric3D is an advanced monocular depth prediction model that processes a single image to estimate the depth of each pixel. Key characteristics of Metric3D include:

- **Training**: It is trained in a supervised manner on a dataset comprising tens of millions of images, akin to the well-known DepthAnything model, ensuring robust generalization capabilities.
- **Scale Accuracy**: When provided with camera parameters or after minor fine-tuning, Metric3D can accurately reconstruct the environment to the correct scale.
- **Surface Accuracy**: Enhanced with an additional loss function focused on surface normals, Metric3D offers improved recognition of object shapes.

Screencast.from.2024.05.28.17.44.17.webm

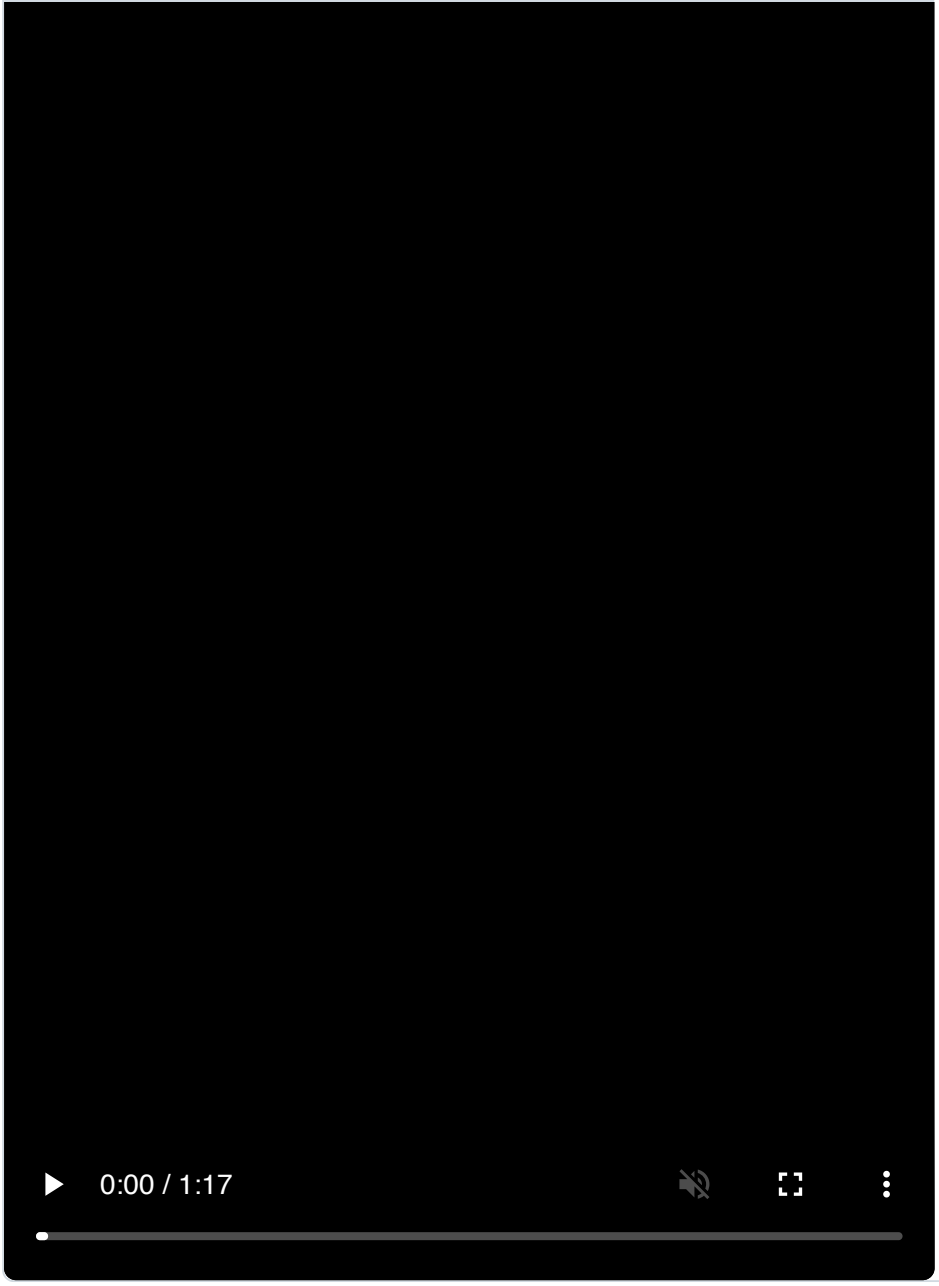
Category

Show and tell

Labels

None yet

2 participants



Exporting to ONNX and ROS2

Our contributions to the community include code for converting Metric3D's pretrained models to the ONNX format, facilitating integration with different systems. The code can be accessed here: YvanYin/Metric3D#102

Additionally, we developed a ROS2 module for processing images into point clouds, which are then aligned with the robot's base_link for use in Autoware: https://github.com/Owen-Liuyuxuan/ros2_vision_inference

Performance Benchmarks

We evaluated the model's inference time across various platforms at its native resolution (616 x 1064).

> There are tests with lower resolutions YvanYin/Metric3D#103

Machine	Setup	Inference Time
GTX 3060 Laptop	TensorRT FP16	0.06s
GTX 4090 Desktop	GPU	0.06s

Machine	Setup	Inference Time
Jetson Xavier	TensorRT FP16	0.5 s

Despite acceptable performance on higher-end GPUs, the model struggles on less powerful devices like the Jetson Xavier without further optimization. But we can conduct the following test with the robot running on my laptop.

Modifications to Autoware

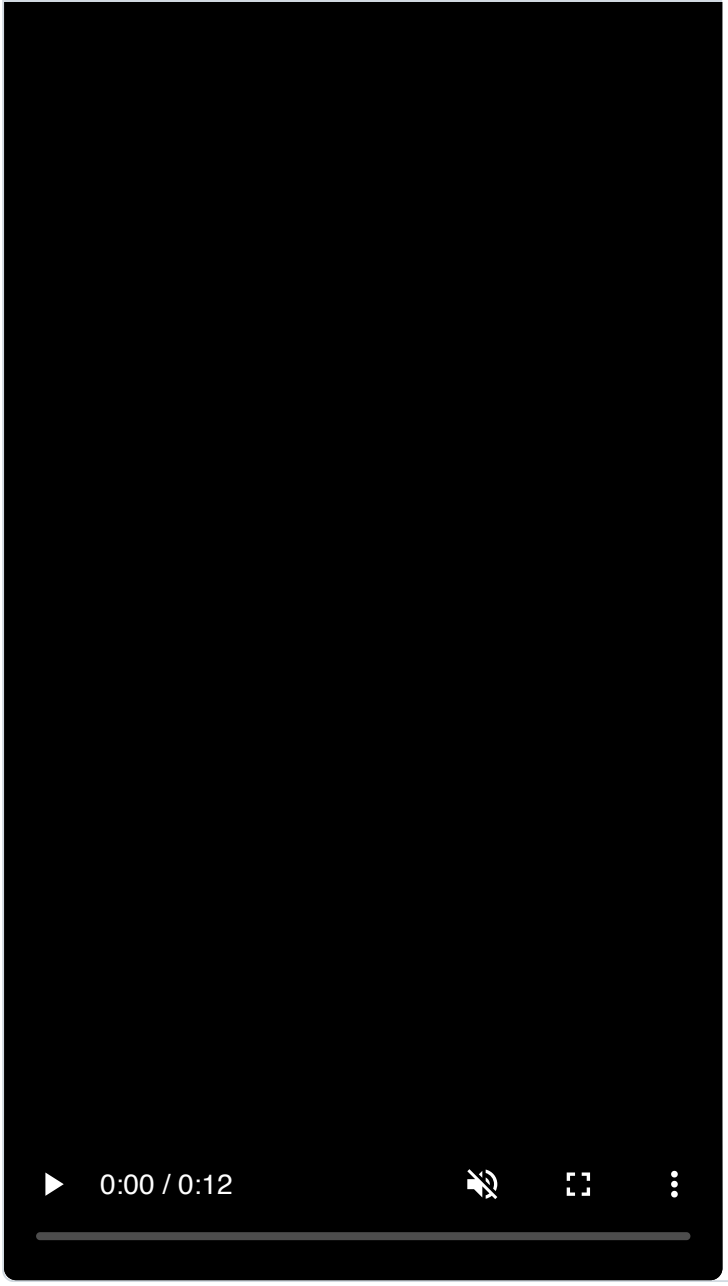
Modifications were necessary within the Autoware framework to accommodate the unique inputs from Metric3D:

- We redirected the point cloud input topic in `tier4_perception_launch/launch/perception.launch.xml` to accept data from Metric3D.
- Adjustments were made to the ground-segmentation and occupancy grid map parameters to ensure accurate environmental representation.
- The localization was not changed and was conducted with LiDAR. The planning module was not changed either.

Test on Indoor Robot

Our preliminary tests with an indoor robot have provided valuable insights, helping us identify areas for enhancement.

VID_20240530_143957.mp4



Future Directions

Looking forward, we aim to refine our approach by:

- Enhancing edge detection through instance segmentation, improving the precision of depth predictions at object boundaries.
- Exploring the integration of monocular 3D detection algorithms to bypass additional point cloud-based detection methods.
- Continuing the calibration and fine-tuning of the Metric3D model for optimal performance across various indoor settings.
- Utilizing the current codes on multi-camera setups to create full-view point clouds.
- Optimizing the processing of ultra-dense point clouds near the camera to enhance downstream performance.

Special Thanks

Special thanks to Metric3D for the great work.

1 comment · 1 reply

Oldest | Newest | Top

DevankGarg on Jun 10

Hello

I am also working on autoware and I have just started working on autonomous vehicles.
Can you please guide and help me get started? It will be highly beneficial for me

I'd very grateful

↑ 2 1 reply

Owen-Liuyuxuan on Jun 10 Collaborator Author

The official documentation is helpful and I will provide some important references for you.

1. If you are installing and testing basic functions on your own PC: https://autowarefoundation.github.io/autoware-documentation/main/installation/; https://autowarefoundation.github.io/autoware-documentation/main/tutorials/ad-hoc-simulation/planning-simulation/
2. If you are considering integrating Autoware into your vehicle: https://autowarefoundation.github.io/autoware-documentation/main/how-to-guides/integrating-autoware/overview/

If you have detailed problems you could raise an issue in the Autoware repositories or in the Discussion page.