Semantic Mapping from RGB-D Images for Robotic Vision Scene Understanding

Ryan Amaudruz *Dorian Bekaert* *Milena Kapralova* *Bogdan Palfi* *Darie Petcu* *Alexandru Turcu*

Abstract

- Captured RGB-D images are converted to point clouds and stitched together using fast global registration.
- A 3D object detector identifies the objects and places bounding boxes around them.

RVSU challenge[1]

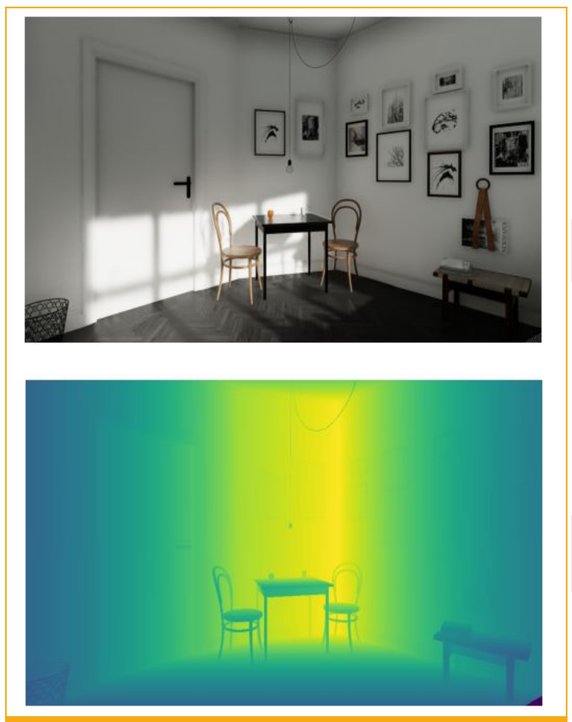
- A simulation where a robotic agent actively explores an indoor environment.
- The goal is to produce 3D bounding boxes with labels around all objects of interest. The result is a 3D semantic map.

Motivation

- Important for robots to understand a dynamic environment and be able to identify and localize objects.
- Important for close human-robot interaction.

Visual SLAM

Simultaneous **L**ocalization **A**nd **M**apping is a technique used in computer vision, where a robot explores an unknown environment with the purpose of building a map of its environment while simultaneously keeping track of its own movement and location. **Visual SLAM** (VSLAM) performs this technique while solely using visual information. In our case, this information is obtained from an RGB-D camera. In **Semantic VSLAM**, just as VSLAM, the agent obtains geometric information about the environment and its movement, but it *also* detects and identifies objects in this environment. This way, the agent has a better understanding of the its environment, allowing the robot to potentially solve more complex reasoning tasks.

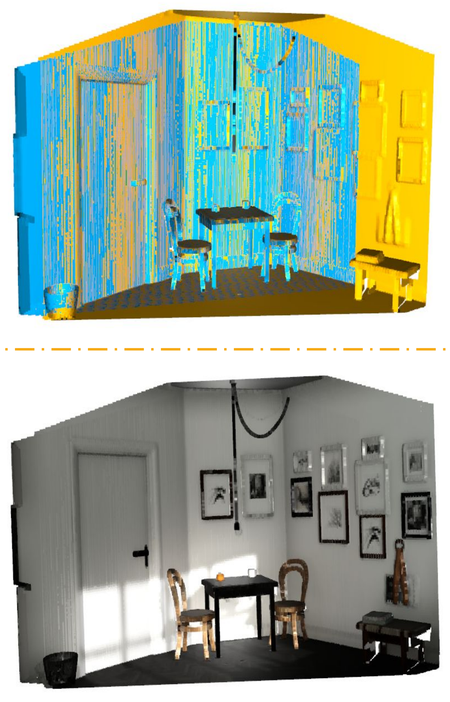


RGB (top) and depth (bottom) image

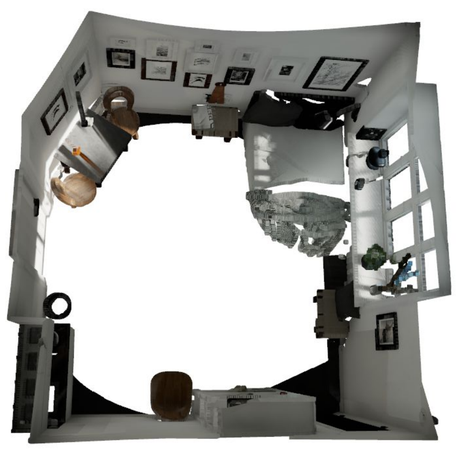
Combine **RGB** and **depth** image to create a point cloud

Point clouds from two different images

Use **Fast Global Registration** to combine point clouds

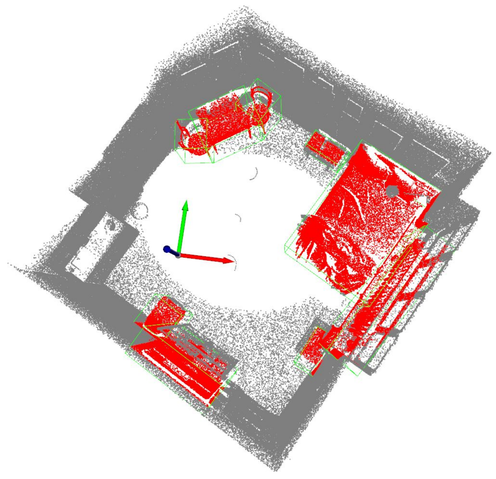


Registration process (top) and resulting point cloud (bottom)

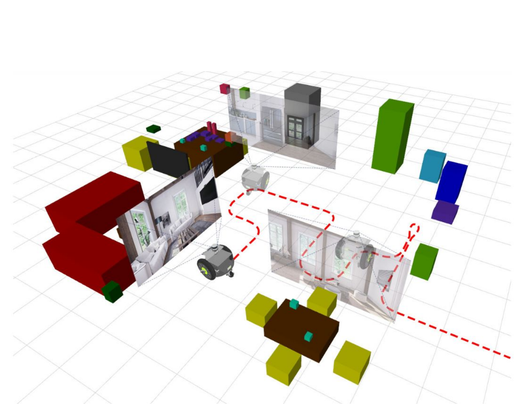


Global point cloud of entire environment

Perform **3D object detection** on the global point cloud



Object detection places bounding boxes around objects (ours)



3D Semantic map example (challenge paper)

Methods & Results

Global Registration

The **fast global registration algorithm**[2] from Open3D takes a source point cloud and finds a **rigid body transformation** to match it with a target point cloud. We first extract 33-dimensional geometric features for each point in both point clouds, which are then used by the algorithm for feature matching. Our pipeline uses the fast global registration algorithm **iteratively** to create the global point cloud in real time.

3D Object Detection

The **MMDetection3D model** from OpenMMLab[3] is used for the 3D object detection task. Given the global point cloud, the model is able to detect bounding boxes around objects and also identify the corresponding object types, providing semantic labels. The model is able to achieve an **accuracy of ~ 90%** on the labels the model is able to find. Unfortunately the model is **unable to assign most of the labels** in the point cloud.

Limitations

- The object detection model we use can only detect 5 out of 25 classes. This can be improved via transfer learning.
- The object detection is only as good as the generated point cloud, whose completeness highly depends on the route the robot takes.
- The robot currently follows a carefully hand-crafted path which has a significant influence on the quality of the map.

Future work

- Future work could explore deep global registration for creating the point clouds or use a classic SLAM algorithm to create the semantic map.
- A 2D segmentation algorithm could be applied on the images before creating the point clouds, to help detect additional classes.
- The robot could process the video directly, instead of separate frames at each step. This would give it the ability to handle dynamic changes in the environment.

References

1. *David Hall, et al. "The Robotic Vision Scene Understanding Challenge." (2020).*
2. *Q.-Y. Zhou, J. Park, and V. Koltun, Fast Global Registration, ECCV (2016).*
3. *Chen, Kai et al. "MMDetection: Open MMLab Detection Toolbox and Benchmark". arXiv preprint arXiv:1906.07155. (2019).*

UvA

Master Artificial Intelligence
Computer Vision 2 course 2023
Group 4

https://github.com/a-turcu/vslam-attempts