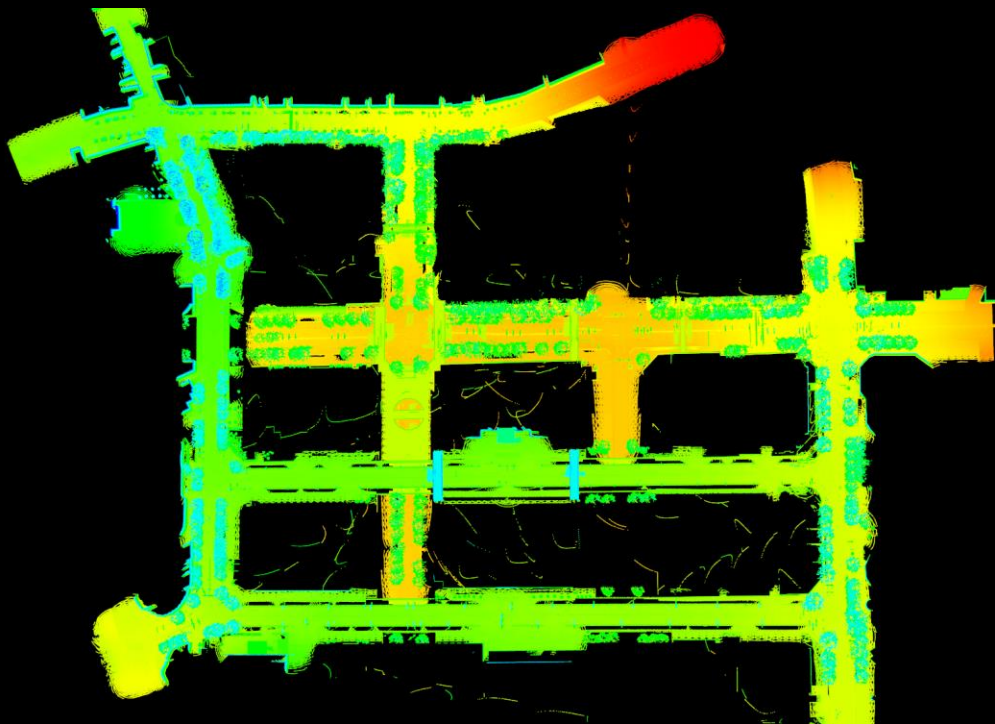ROBOTEC AI

Piotr Rybicki

ROSCON
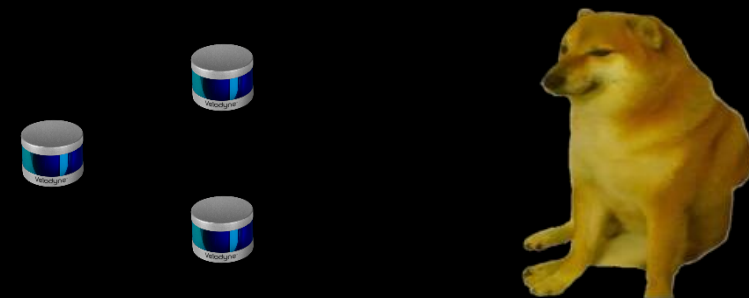KYOTO
2022

Robotec GPU Lidar

Performance

350 MLN rays/second

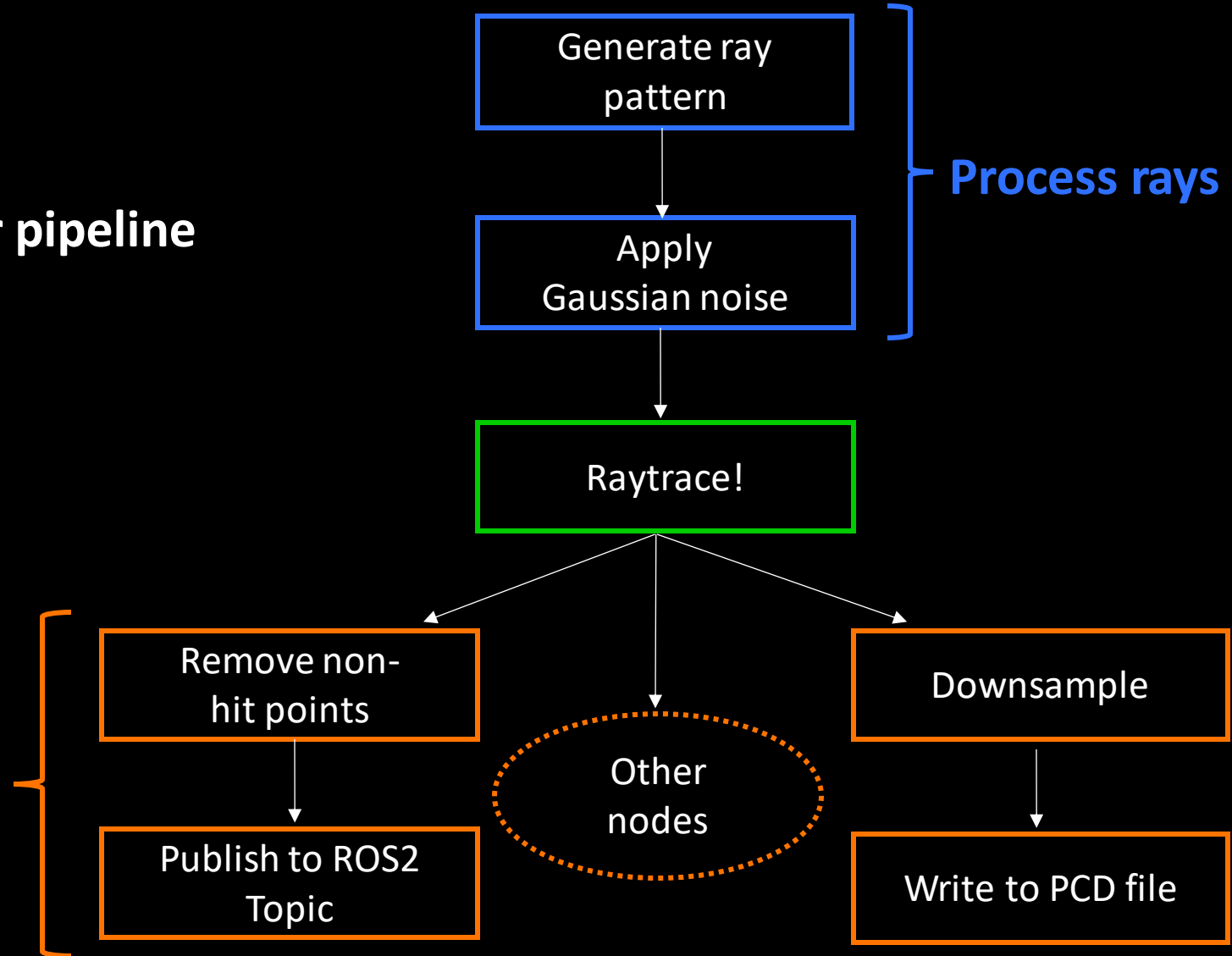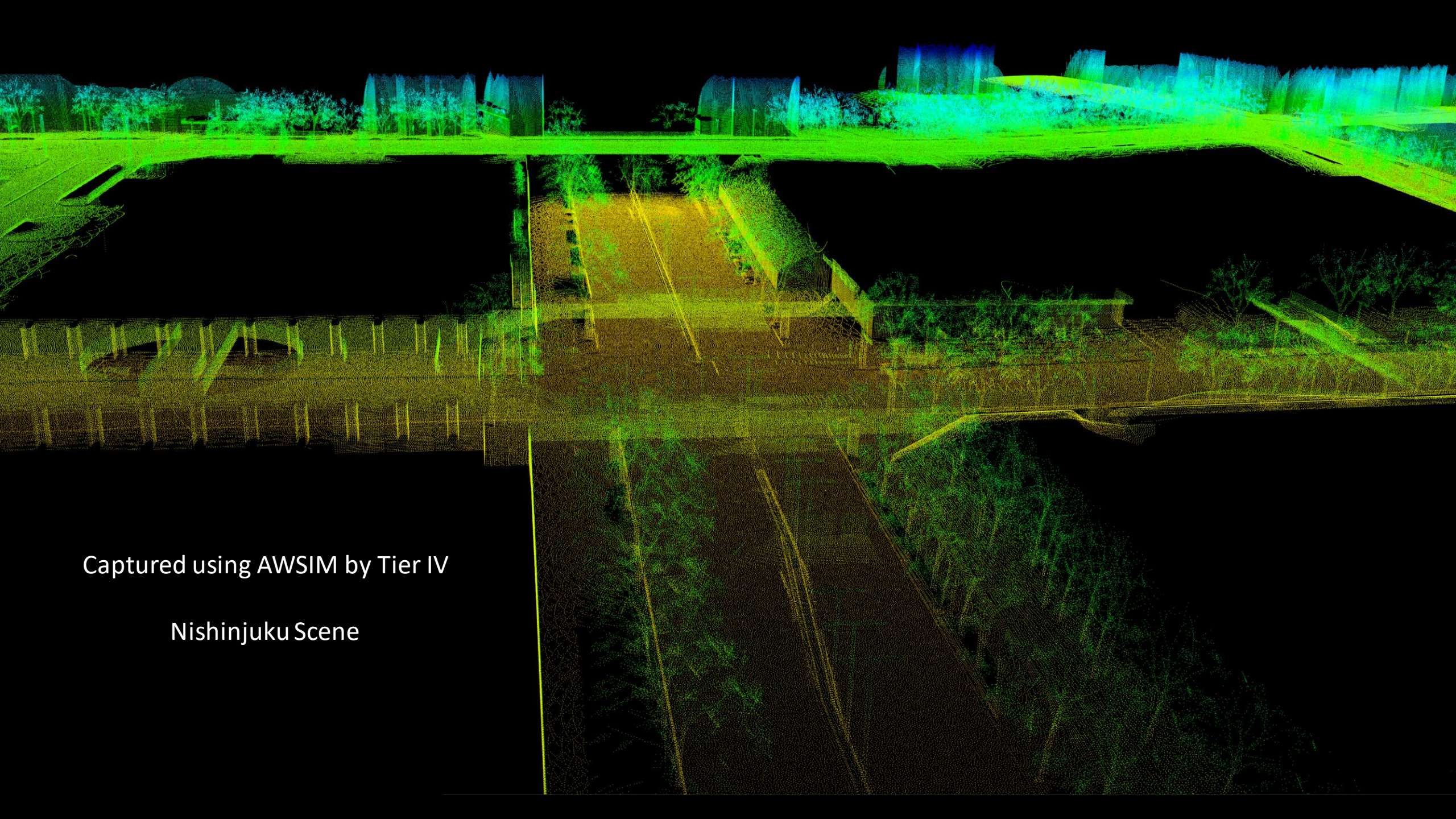On a complex urban scene (below)

CPU multi-threaded solution

RTX 3070 Laptop

Flexiblity

Dynamically define your pipeline

- Non-periodic patterns
- Downsampling
- Writing to PCD file
- (check docs for more!)

```
┌──────────────┐
│ Generate ray │ ─┐
│   pattern    │  │  Process rays
└──────┬───────┘  │
       ▼          │
┌──────────────┐  │
│    Apply     │  │
│Gaussian noise│ ─┘
└──────┬───────┘
       ▼
┌──────────────┐
│   Raytrace!  │
└──────────────┘
```

Process rays

Process pointcloud

```
┌──────────────┐                      ┌──────────────┐
│ Remove non-  │      ( Other  )      │  Downsample  │
│  hit points  │      ( nodes  )      └──────┬───────┘
└──────┬───────┘                             ▼
       ▼                              ┌──────────────┐
┌──────────────┐                      │Write to PCD  │
│Publish to ROS2│                     │    file      │
│    Topic     │                      └──────────────┘
└──────────────┘
```
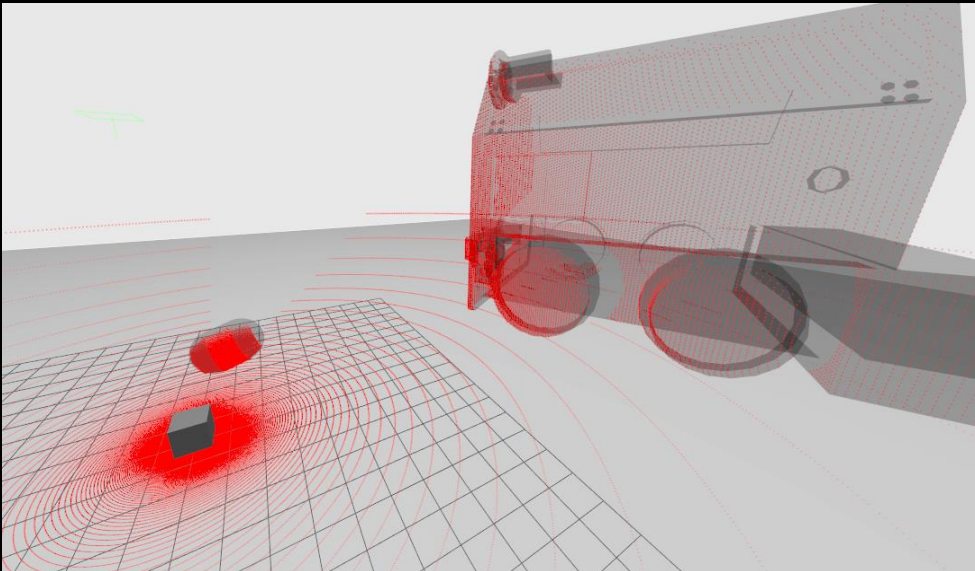
Captured using AWSIM by Tier IV

Nishinjuku Scene

Easy to use

- C-API available from any language

- Integrated with AWSIM (Tier IV) (C#, Unity)
- Gazebo Plugin in progress
- Integration ideas are welcomed!

```
RGL_API rgl_status_t
rgl_entity_create(rgl_entity_t *out_entity, rgl_scene_t scene, rgl_mesh_t mesh);

RGL_API rgl_status_t
rgl_entity_set_pose(rgl_entity_t entity, rgl_mat3x4f *local_to_world_tf);

/* ... */

RGL_API rgl_status_t
rgl_graph_get_result_data(rgl_node_t node, rgl_field_t field, void* data);
```

Takeaway

https://github.com/RobotecAI/RobotecGPULidar
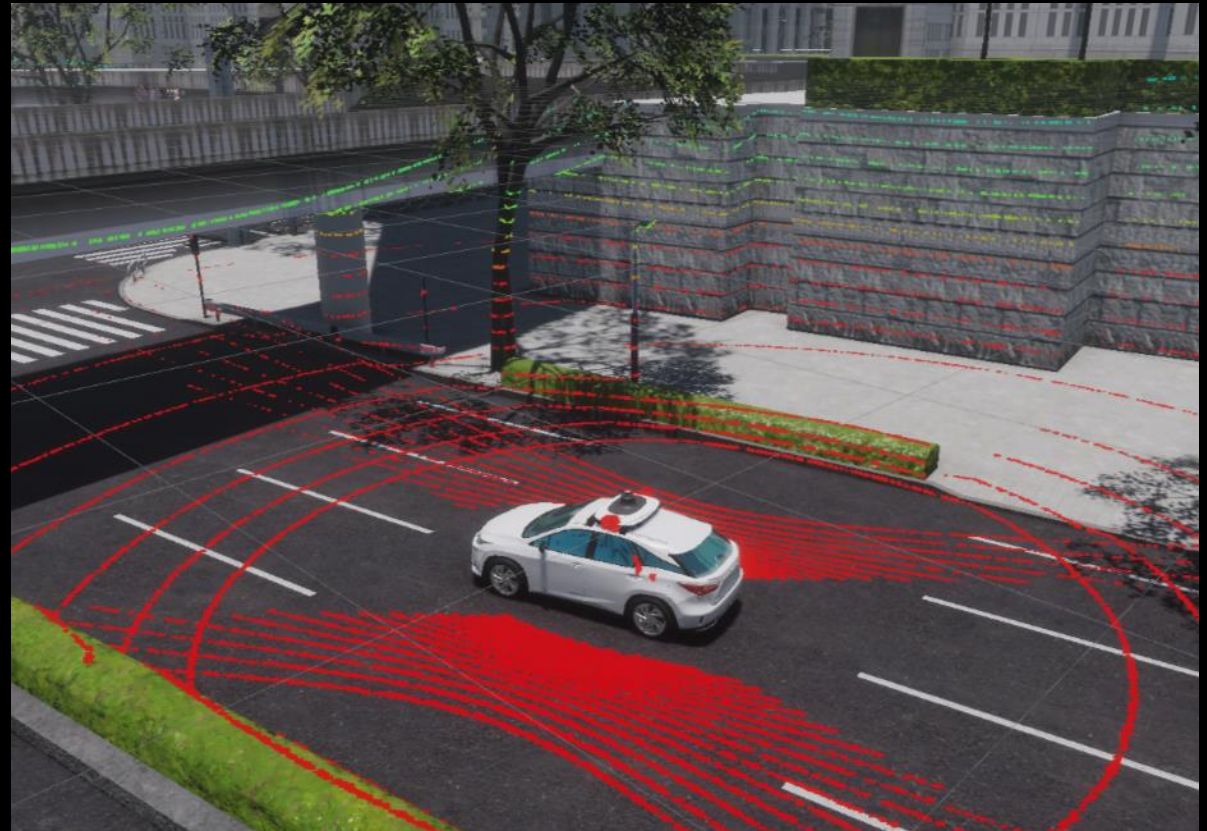
Visit our booth to see cool videos!

ROBOTEC AI