HOCHSCHULE
RAVENSBURG-WEINGARTEN
UNIVERSITY
OF APPLIED SCIENCES

LIDAR AND RADAR SYSTEMS
GRADED HANDS-ON PROJECTS

Prof. Dr. Stefan Elser

GENERAL RULES AND INFORMATION

Graded Hands-On Project
The report

In these slides, you will get a collection of all information needed to solve the tasks of the Hands-On projects.

For each task, you will have to **hand in a report via Moodle with a strict deadline.** In this report you will have to answer all questions defined in the task descriptions.

All limitations of your reports (e.g. page-count) will also be published via Moodle.

Graded Hands-On Project
The report

A report should always carry all information for the reader to reproduce your algorithms to get the same results as you. At least provide always the following:

- **What?**

  What is your answer or result to the question?

- **How?**

  How did you come to your conclusion? How did you calculate your result?

- **Why?**

  Why did you use your approach? Why did you define the function the way you did it?

Graded Hands-On Project
The report

A report should always carry all information for the reader to reproduce your algorithms to get the same results as you. At least provide always the following:

- **What?**
- **How?**
- **Why?**

Describe your own functions in detail to allow a reproduction by the reader.
Here, pseudo code or a flow chart can be helpful to describe longer or more complex functions.

Graded Hands-On Project
The report

Follow the general rules for a scientific report. Read the “How to cite” guide provided.

Any kind of plagiarism will result in an automatic failing grade (5.0) for all students involved.

Graded Hands-On Project
The report

As a typical What You See Is What You Get (WYSIWYG) editing software (e.g. Microsoft Word, Apple Pages) will reach its limitations for a scientific report, **we will use LaTeX only**.

To generate your report, use one of the many available LaTeX editors (e.g. TeXworks): https://en.wikipedia.org/wiki/Comparison_of_TeX_editors

As you will most likely have to use LaTeX for compiling your Master thesis, this is a good place to start practicing.

Graded Hands-On Project
The report

You can also use your RWU account to access the overleaf hosted by the Verfassten-Studierendenschaft (https://vs.rwu.de/overleaf):

https://overleaf.vs.rwu.de/

If you log in, you can generate a new project using a provided template. This allows a very easy start if you never worked with LaTeX before. But remember:

ALWAYS MAKE SURE TO SAVE YOUR PROGRESS LOCALLY ON YOUR MACHINE!

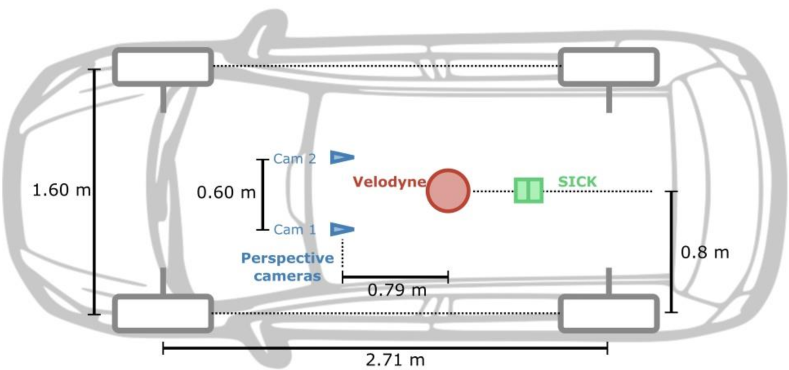
Now, let us talk about what we want to do....

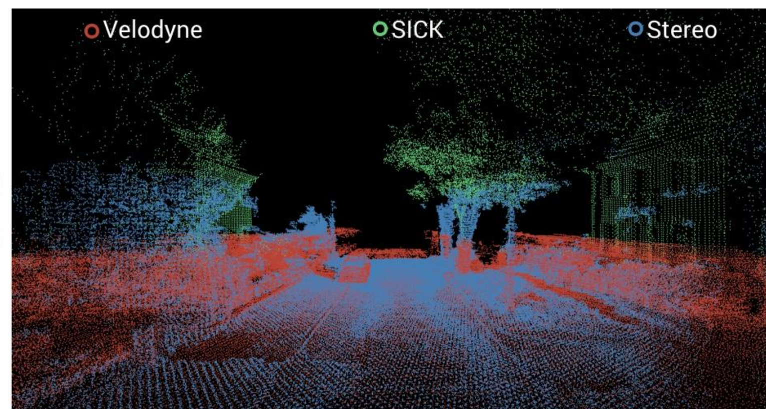
THE PROJECT TASK:
EVALUATION OF AN 3D OBJECT DETECTOR

Evaluation of an 3D object detector
Dataset and algorithm to be used

During this task, we want to understand how we can evaluate the detections in a recorded lidar point cloud.

For this task, we will work with a subset of the KITTI-360 dataset in a reduced format.





Evaluation of an 3D object detector
Dataset and algorithm to be used

The original KITTI dataset as seen on the right did provide 3D bounding boxes.

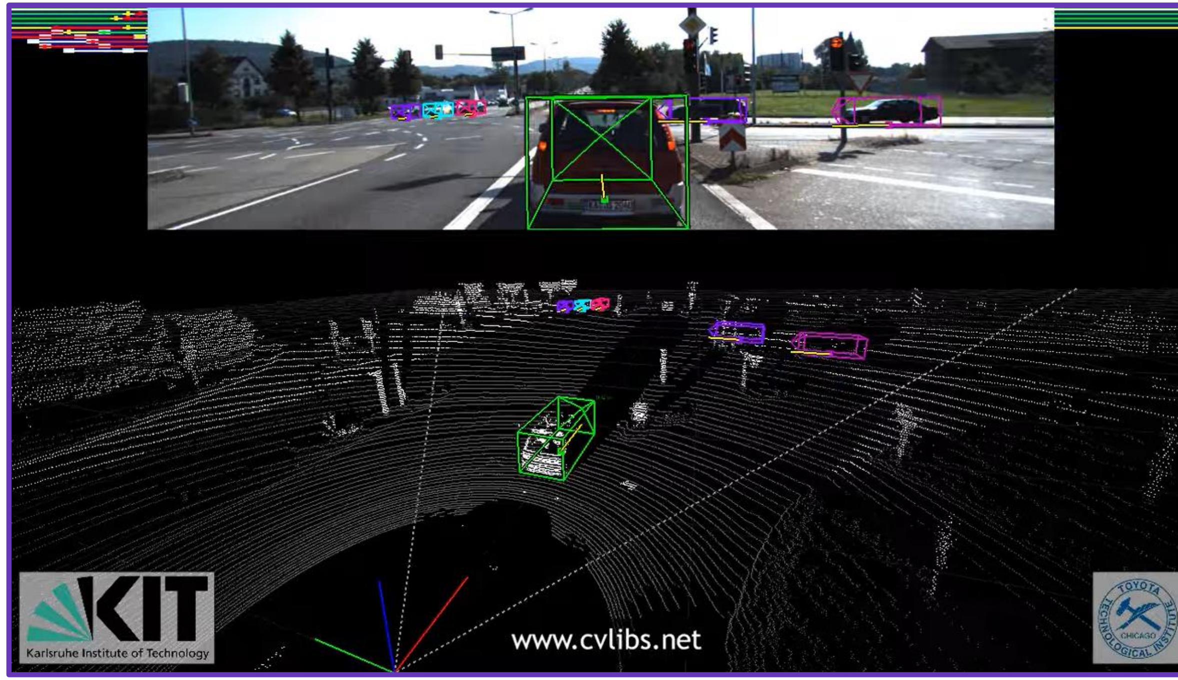


Fig.: Scene with labelled objects from the original KITTI dataset.
Source: https://www.cvlibs.net/datasets/kitti/index.php

Evaluation of an 3D object detector
Dataset and algorithm to be used

The KITTI-360 dataset is an improvement over the original KITTI dataset as it now includes also the instance segementation for both 2D and 3D data.

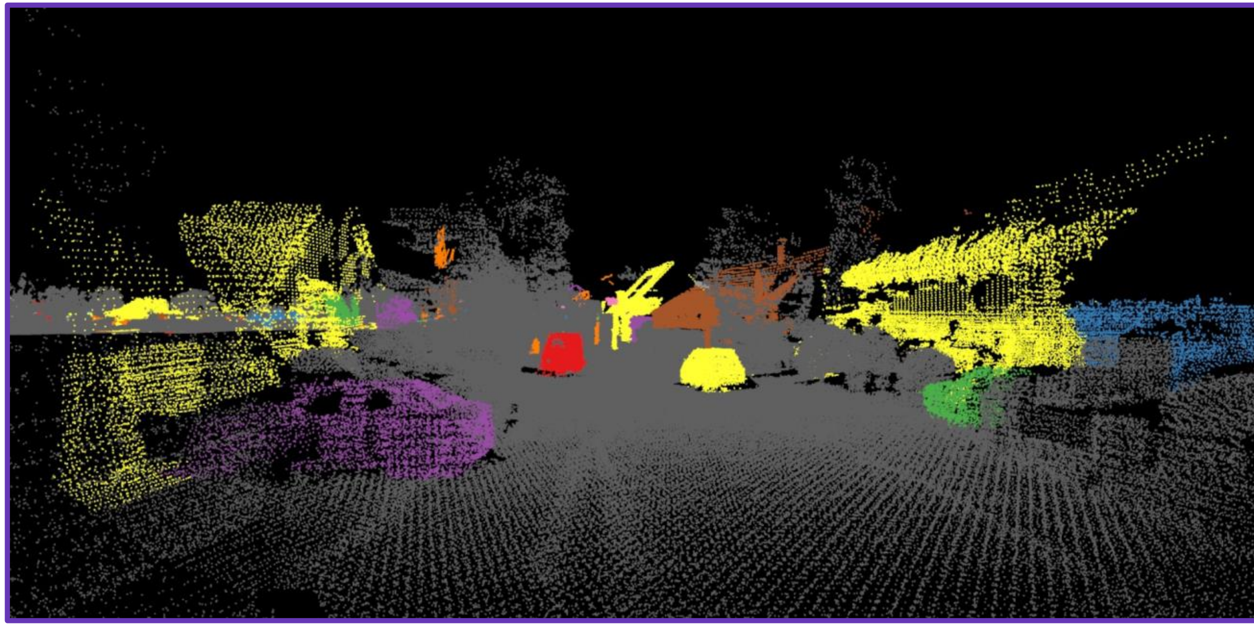
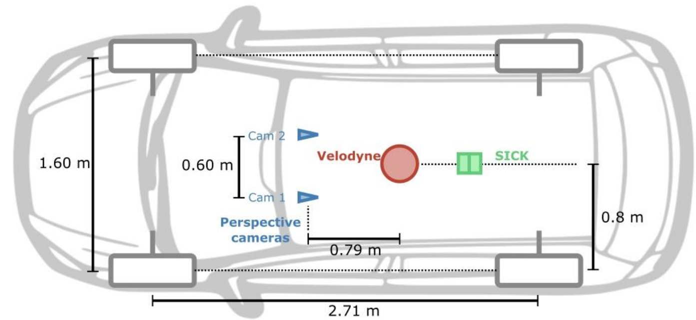
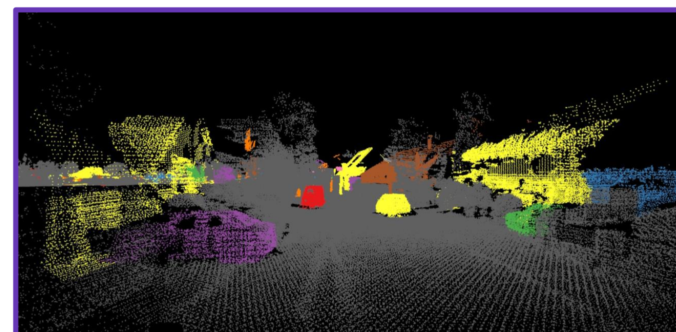


Fig.: Scene with labelled 3D pointcloud from the KITTI-360 dataset.
Source: https://www.cvlibs.net/datasets/kitti-360/

Evaluation of an 3D object detector
Dataset and algorithm to be used

Similar to the Computer Vision project, we also we did select some scenes from the KITTI dataset for easier access. In these scenes, we also reduced the available objects, ground truth data and calibration files.



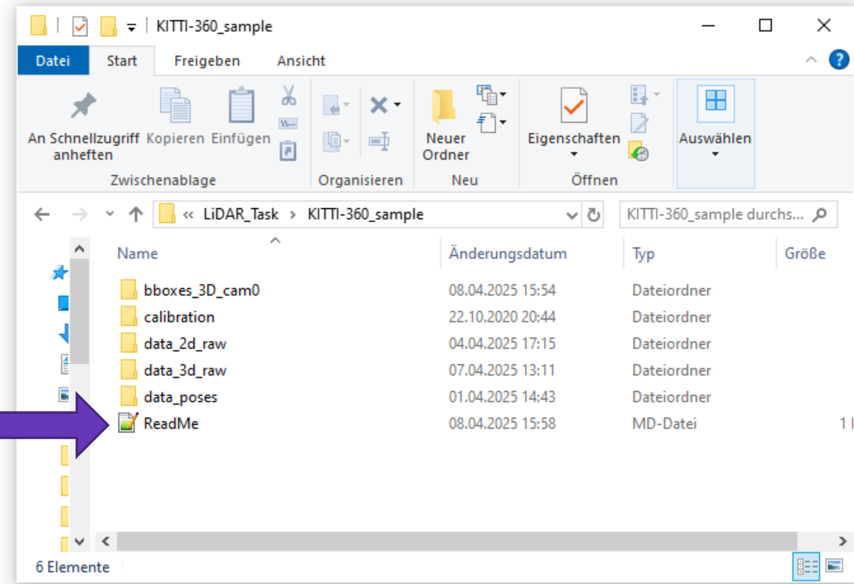


So what is available in our selection?

Evaluation of an 3D object detector
Dataset and algorithm to be used

Fig.: The last file in the folder is the most important file.

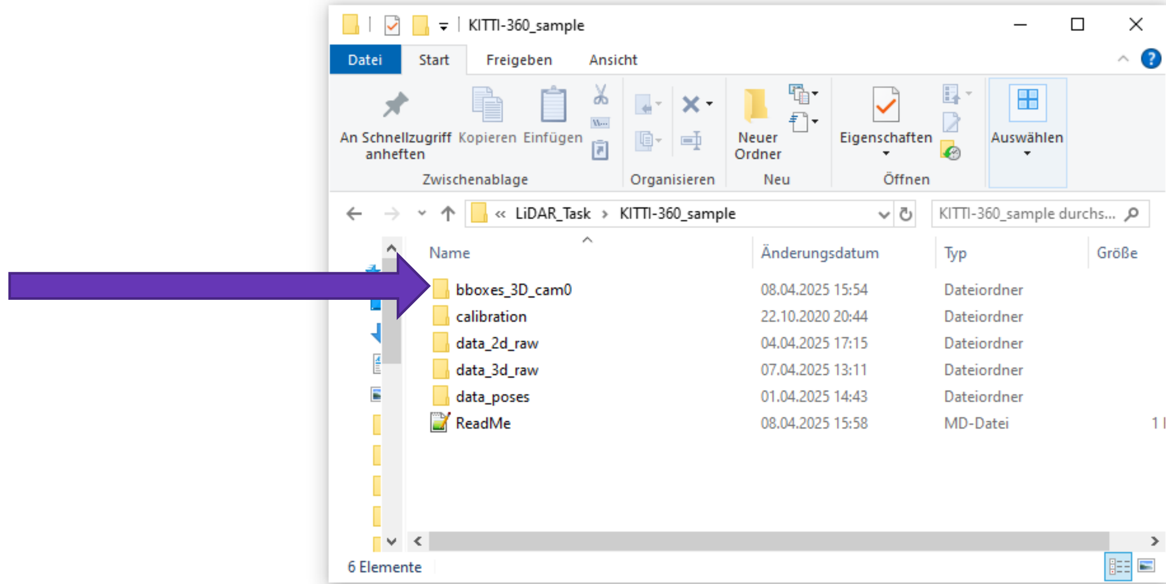
Read the ReadMe file to get additional information how to use the data provide as well as a link to the official github repo from the KITTI-360 team.



Evaluation of an 3D object detector
Dataset and algorithm to be used

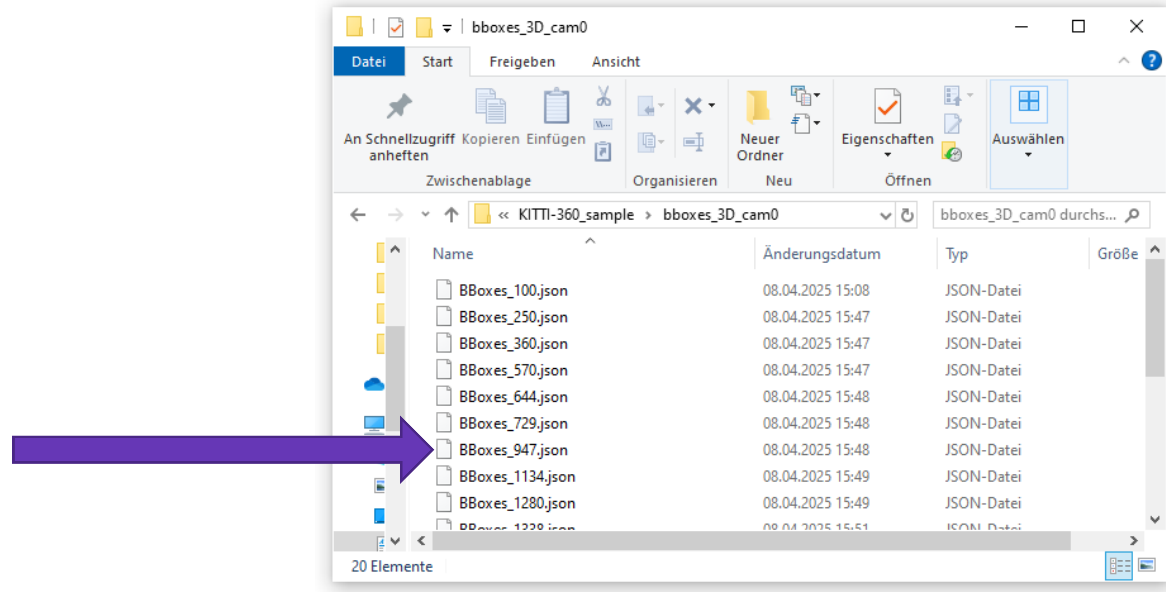
Fig.: The **3D bounding boxes of the cars** in the corresponding scene.

This is not available in the KITTi-360 dataset. We did calculate them to provide an easy access for you.



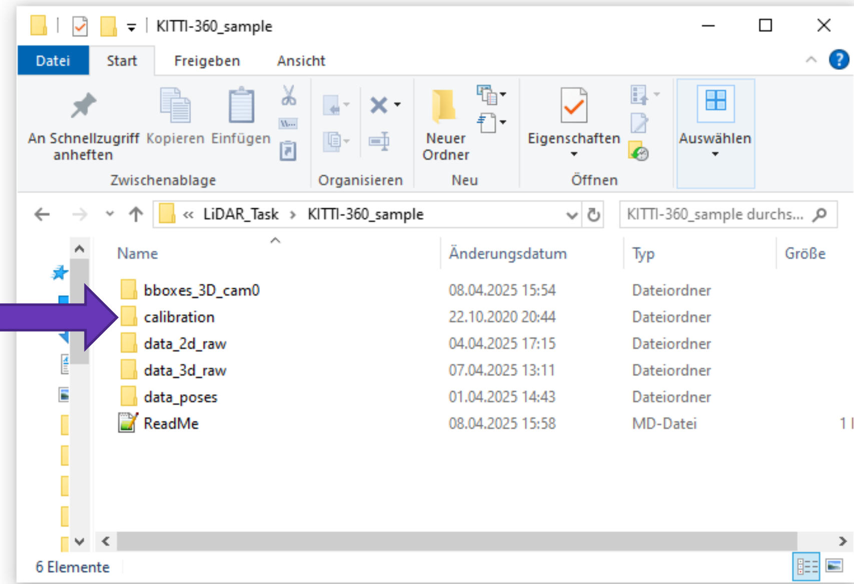
Evaluation of an 3D object detector
Dataset and algorithm to be used

Fig.: All available 3D bounding boxes are provided for each scene. The scene is identified with the the number, e.g. BBoxes_947 does provide a list of all cars available in the scene with ID 0000000947.



Evaluation of an 3D object detector
Dataset and algorithm to be used

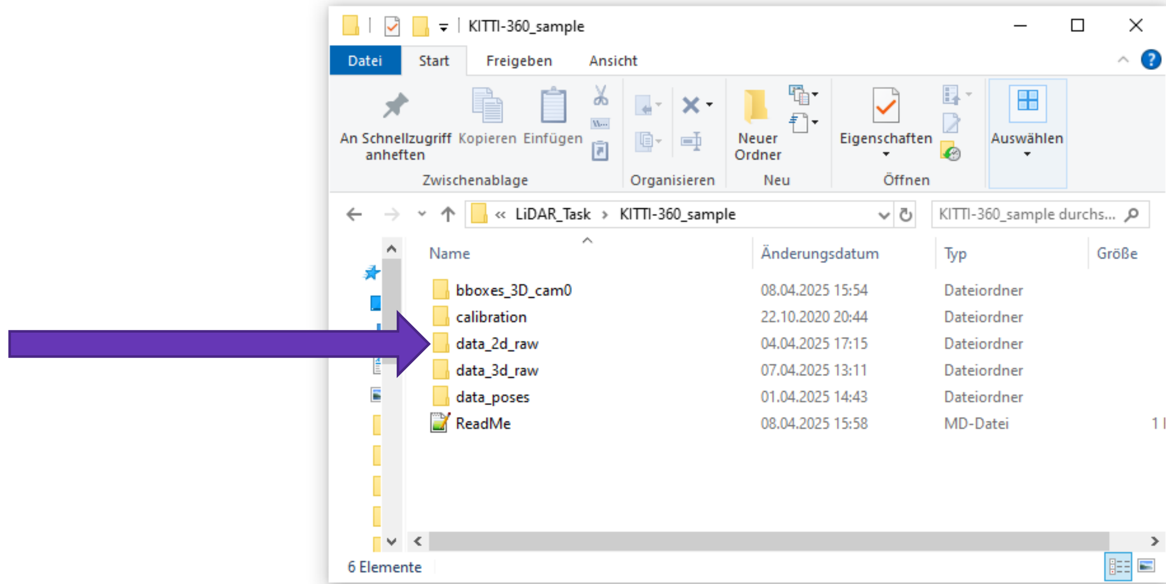
Fig.: The calibration file provides the lidar to camera coorindate transformation as well as the intrinsic matrices for all cameras. This is constant for the full dataset.



We will use the coordinate transformation and not the intrinsics.

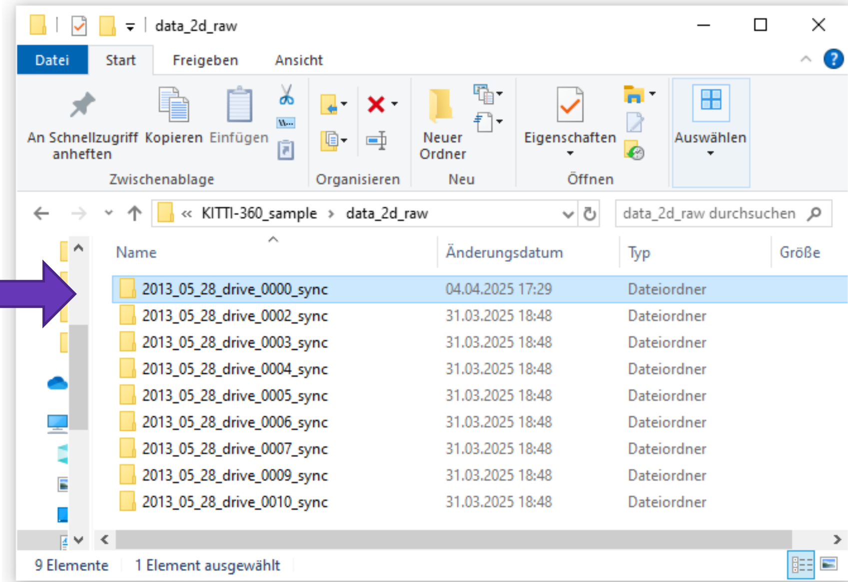
Evaluation of an 3D object detector
Dataset and algorithm to be used

Fig.: The folder with the 2D data includes the camera images. For this task, we will use the left top camera.



Evaluation of an 3D object detector
Dataset and algorithm to be used

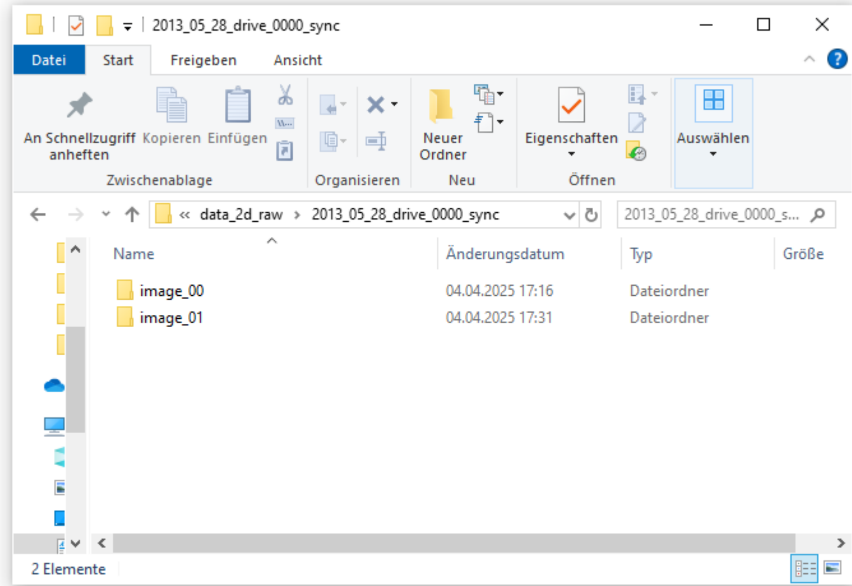
Fig.: If you open the folder, you need to select the first folder again.



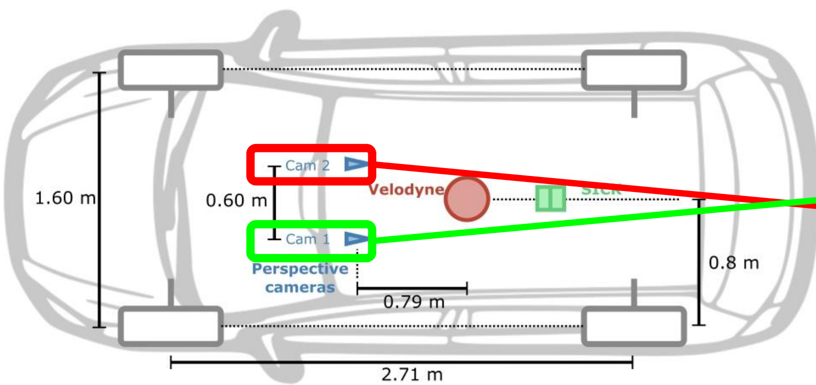
Evaluation of an 3D object detector
Dataset and algorithm to be used

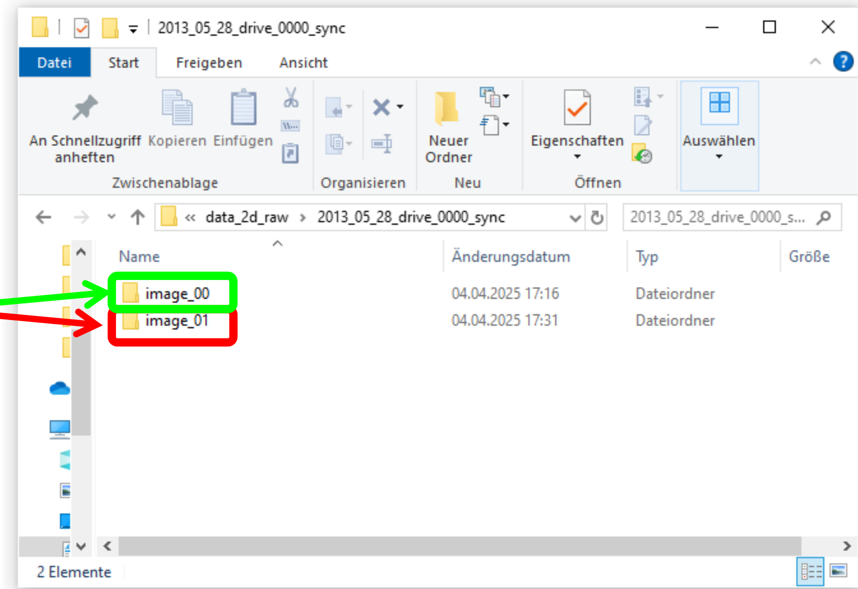
Fig.: The folder image_00 contains the images of the left camera, the folder image_01 contains the images of the right camera.

The naming is a bit confusing compared to the sensor overview.



Evaluation of an 3D object detector
Dataset and algorithm to be used

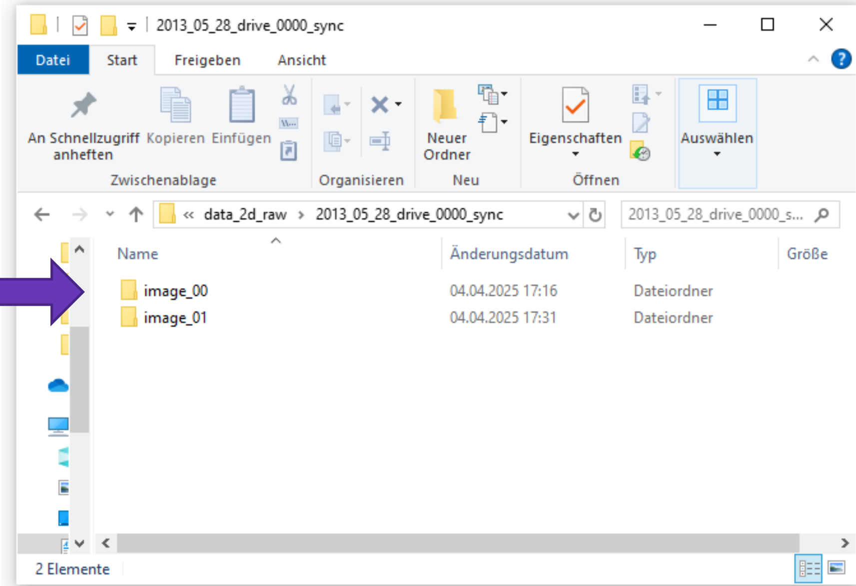




Evaluation of an 3D object detector
Dataset and algorithm to be used

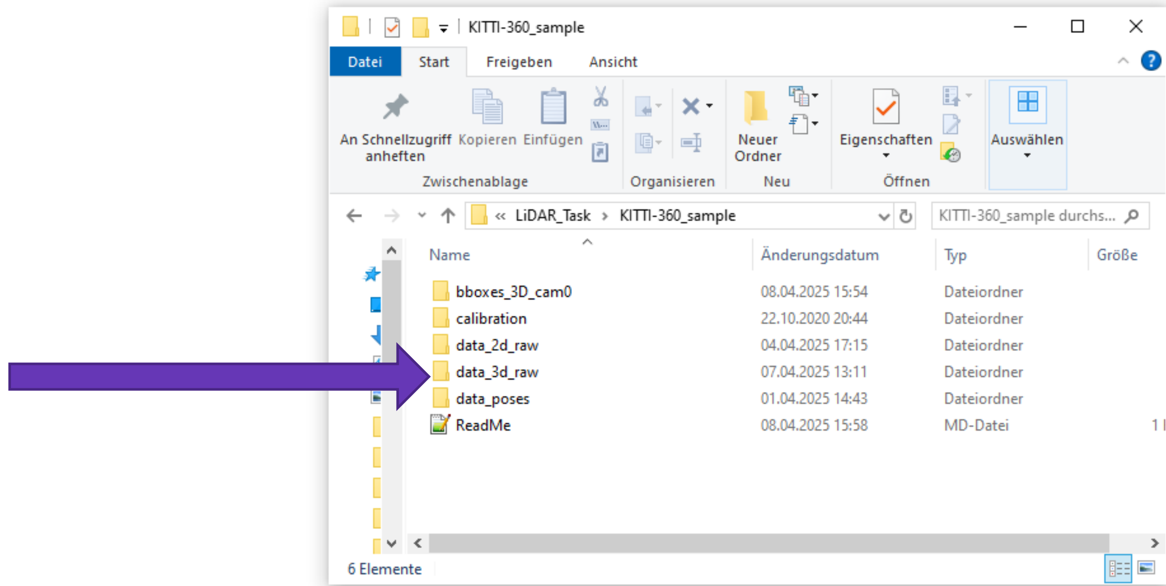
Fig.: For this task, we will use the left top camera in the corresponding folder image_00.

You can take a look at the images right away. You will also recognize the scene IDs mentioned earlier.



Evaluation of an 3D object detector
Dataset and algorithm to be used

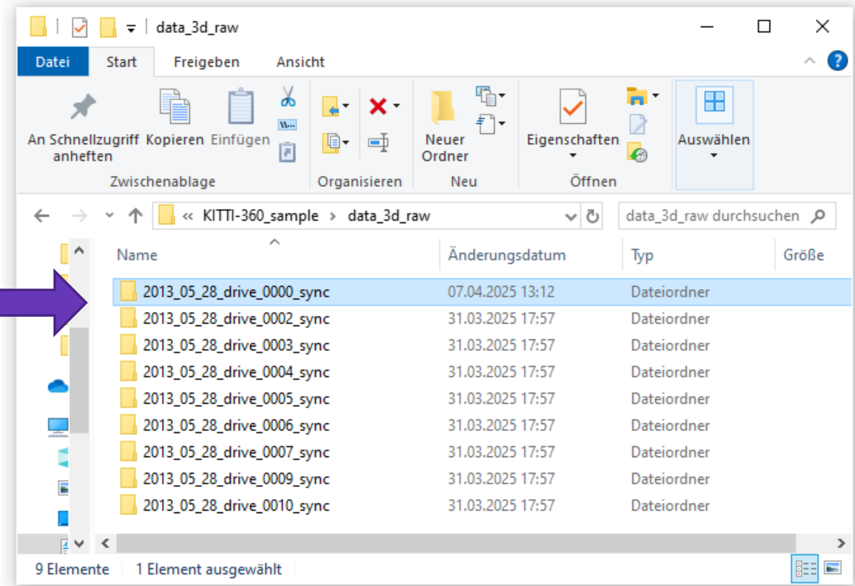
Fig.: The folder with the 3D data includes the lidar measurements. We will use the velodyne data for this task. The structure is the same as in the 2D folder.



Evaluation of an 3D object detector
Dataset and algorithm to be used

Fig.: Again, you need to select the first folder again.

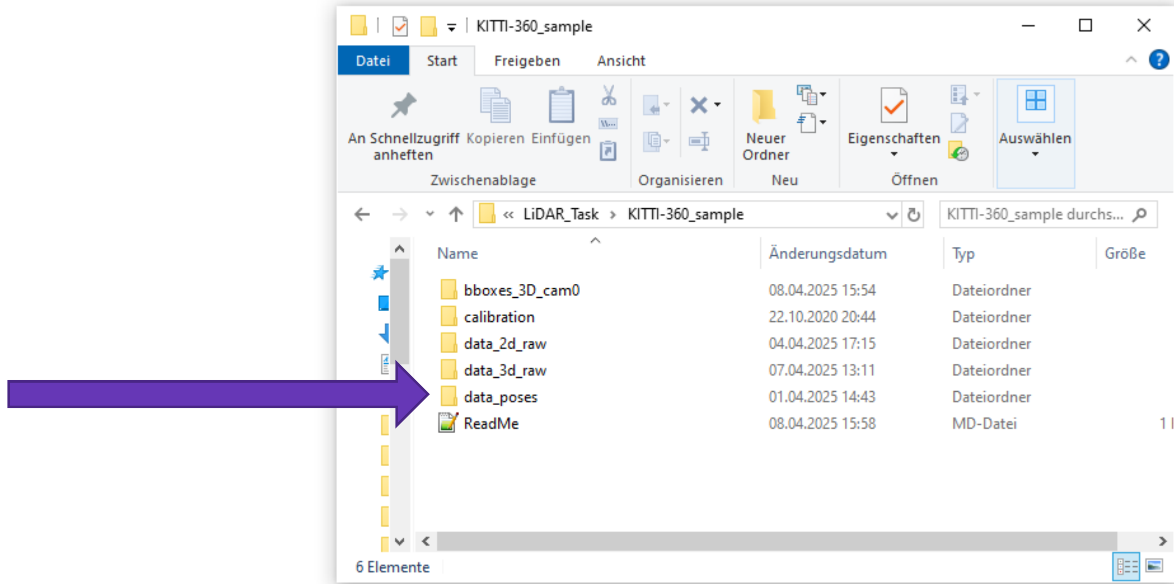
In the subfolder called velodyne_points you will find the pointclouds as bin files again listed with the corresponding ID.



Evaluation of an 3D object detector
Dataset and algorithm to be used

Fig.: The data_poses folder does provide the coordinate transformations for each scene into the world coordinate system.

This is not needed for this task.



Evaluation of an 3D object detector
Dataset and algorithm to be used

Fig.: Now, you should be able to find this scene in the camera folder. Using the helper function, you should also be able to access the corresponding 3D pointcloud.

Evaluation of an 3D object detector
Dataset and algorithm to be used

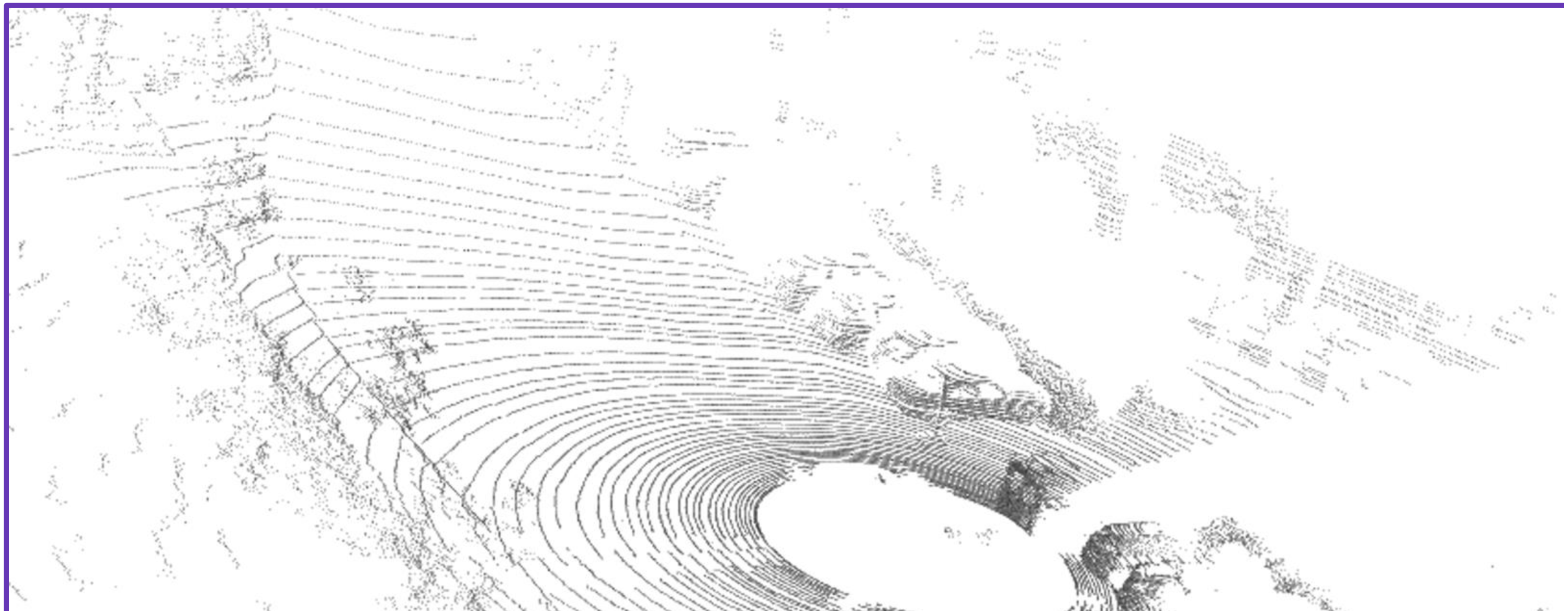
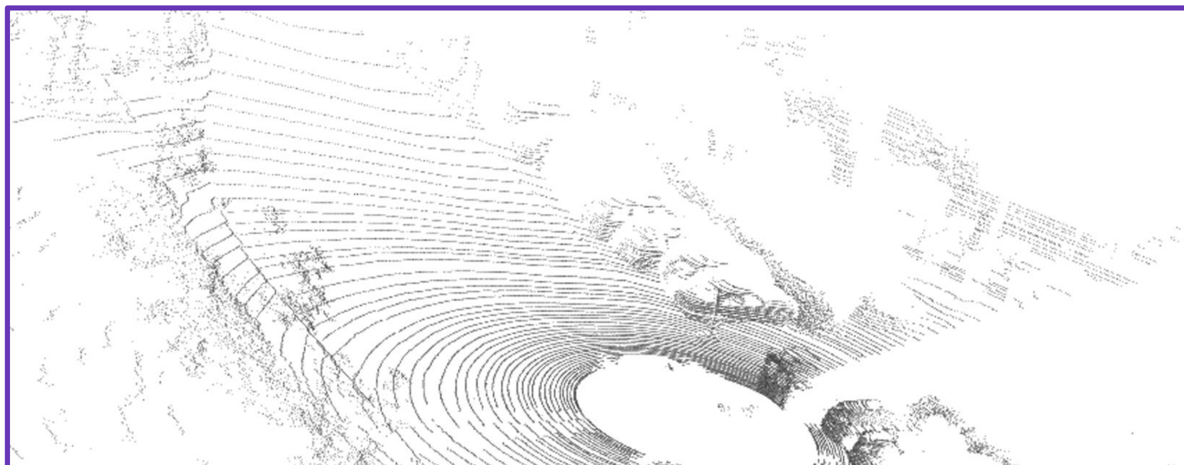


Fig.: The same scene now as 3D pointcloud..

Evaluation of an 3D object detector

Dataset and algorithm to be used



Evaluation of an 3D object detector
Dataset and algorithm to be used

Evaluation of an 3D object detector
What we have to do

We already know how to detect objects in camera images. For this, we again use the ultralytics YOLO but now we use the sematic instance segmentation to detect objects on a pixelwise level.

This can be used quite easily by pip install:

https://pypi.org/project/ultralytics/

https://docs.ultralytics.com/

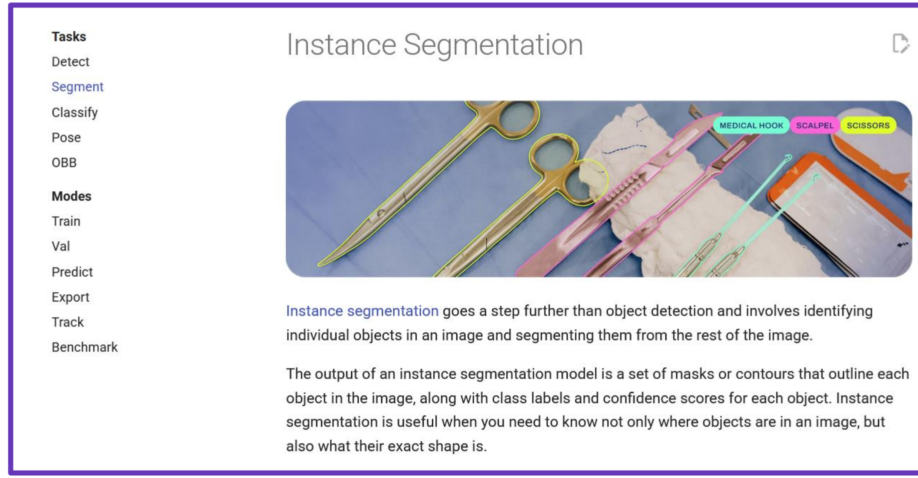


Fig.: Screenshot from https://docs.ultralytics.com/tasks/segment/

Evaluation of an 3D object detector
What we have to do

Fig.: Output of a YOLO semantic instance segmentation. Note that we now have not just the bounding boxes but also the pixelwise labeling available. Now, we can fuse the camera information with the lidar pointcloud.

Evaluation of an 3D object detector
What we have to do

Fig.: For this task, we will focus on the car class exclusively. So you need to get rid of all non-car detections.

Evaluation of an 3D object detector
What we have to do

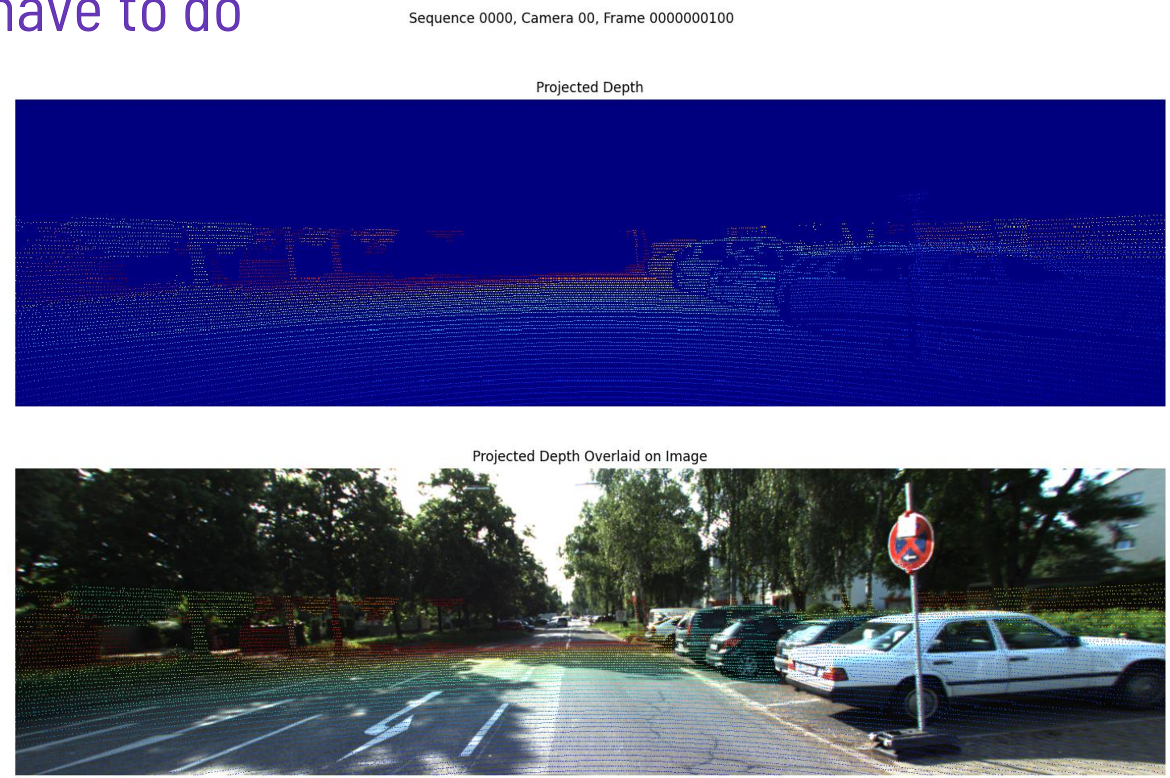
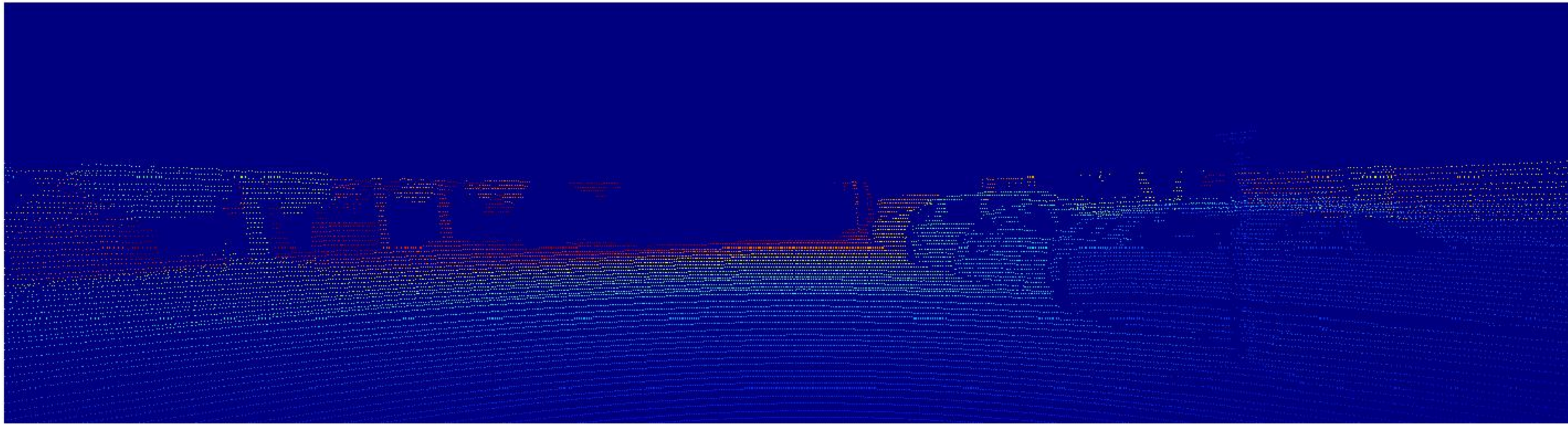


Fig.: Above: Lidar Pointcloud. Below: Camera image with projected points.

Projected Depth



Projected Depth Overlaid on Image

Evaluation of an 3D object detector

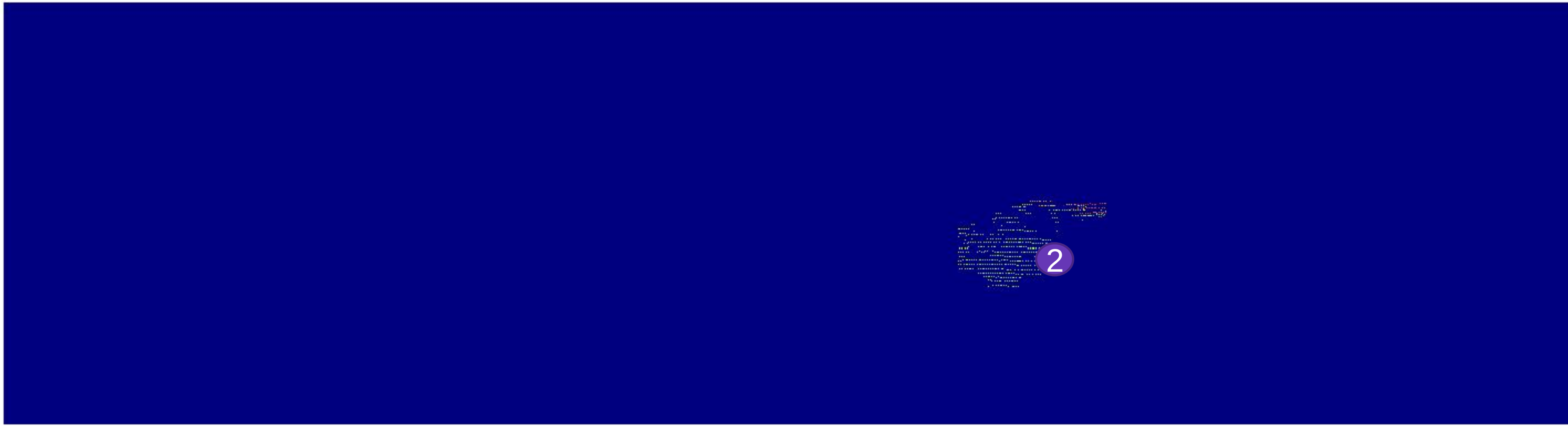
What we have to do

Fig.: Using the masks provided by YOLO, we can now decide which points do belong to what car.

Projected Depth

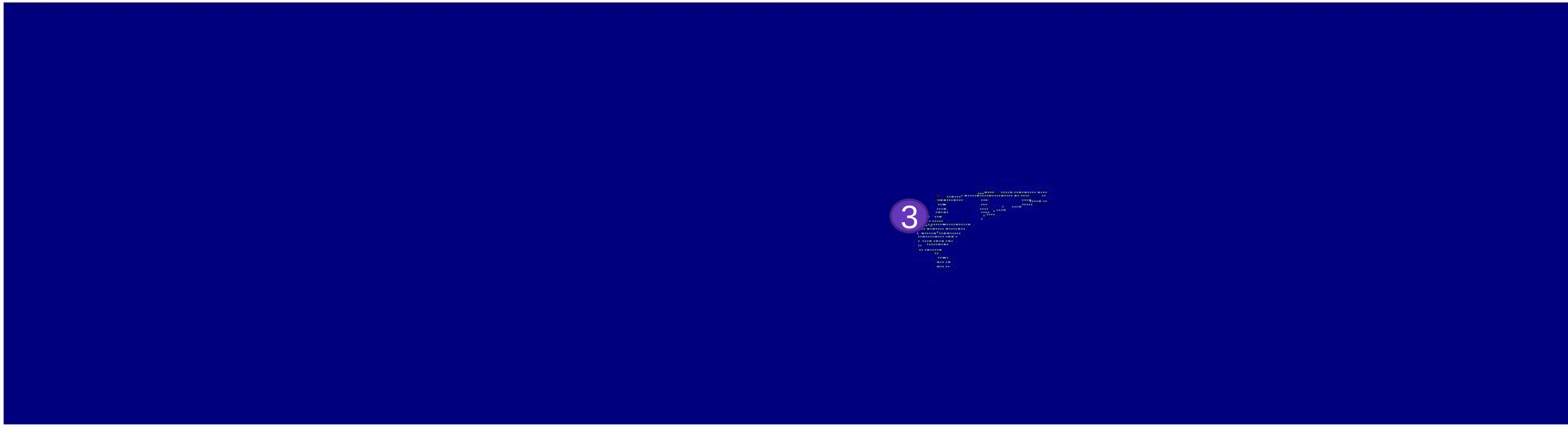
Projected Depth Overlaid on Image

Projected Depth



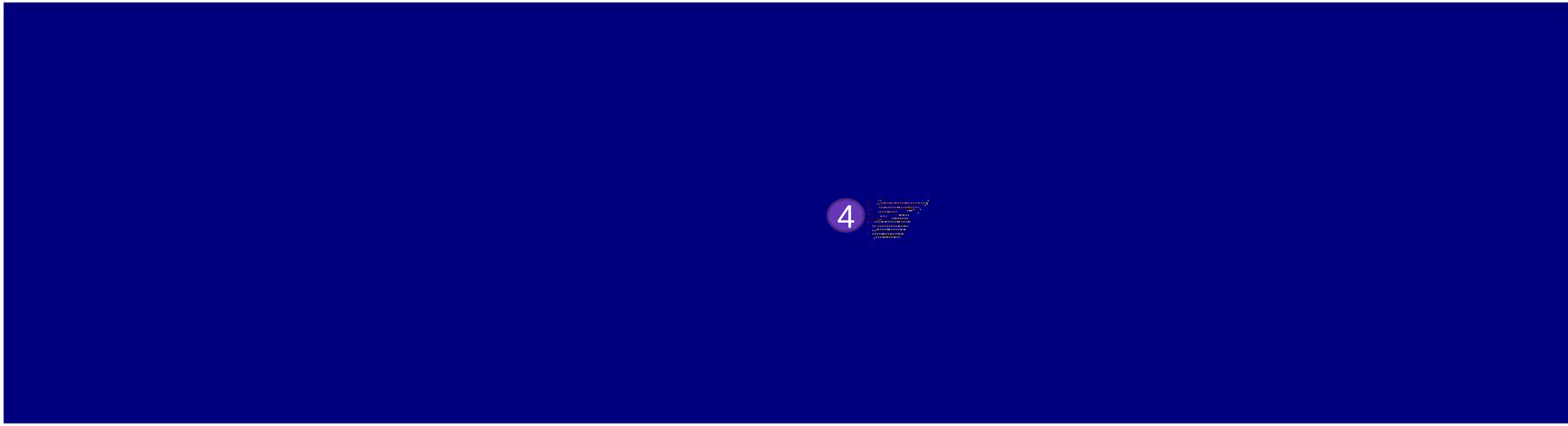
Projected Depth Overlaid on Image

Projected Depth



Projected Depth Overlaid on Image

Projected Depth

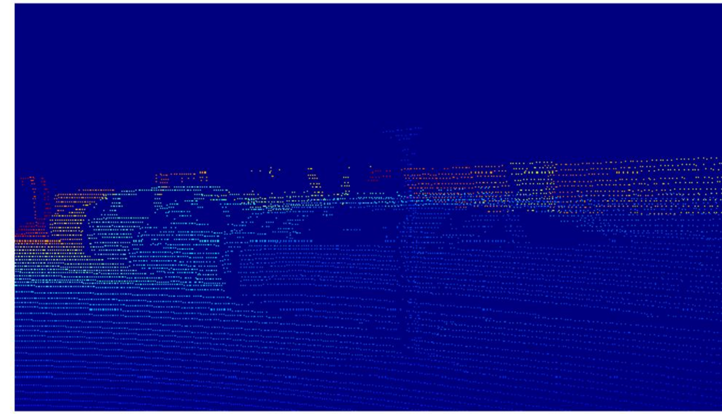


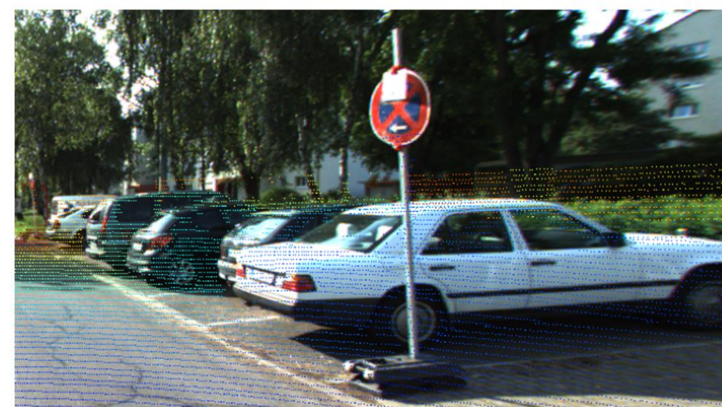
Projected Depth Overlaid on Image

Evaluation of an 3D object detector
What we have to do

So we can use the YOLO output to decide which of the projeced lidar points do belong to what car.

This means, we can also decide which of the original 3D points do belong to what object.





Evaluation of an 3D object detector
What we have to do

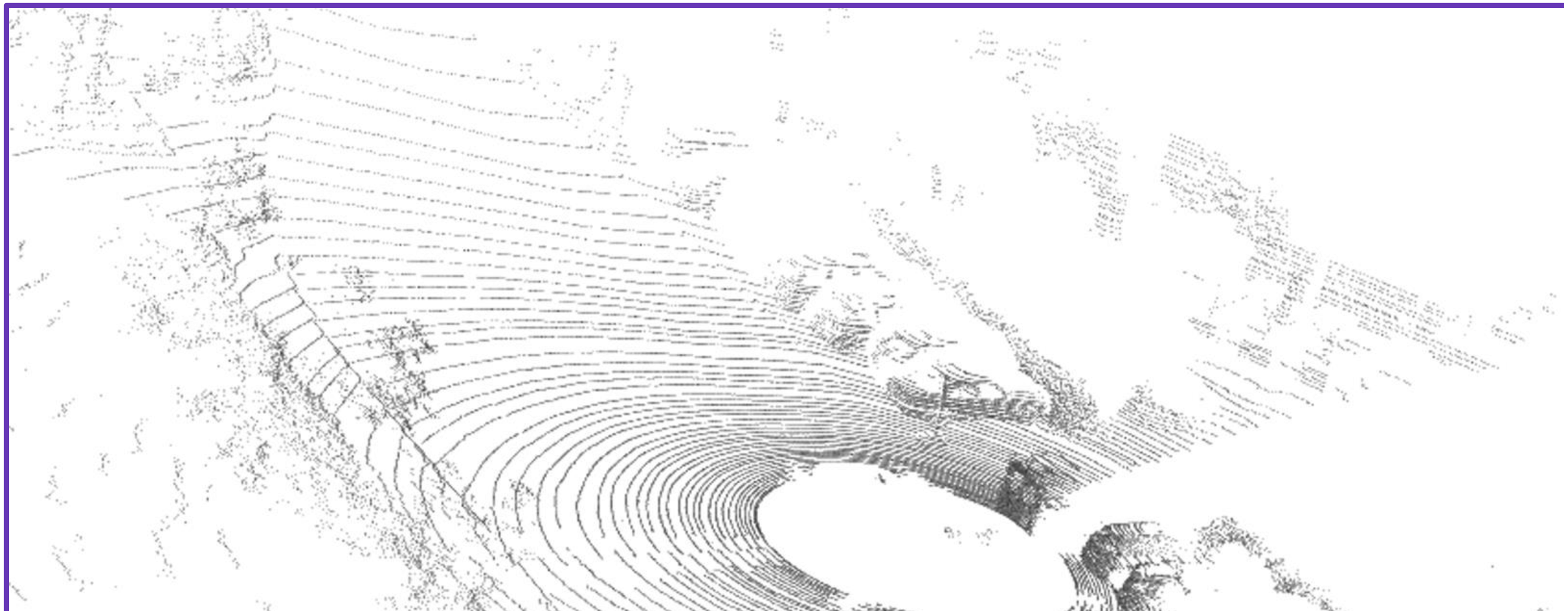


Fig.: The same scene as 3D pointcloud.

Evaluation of an 3D object detector
What we have to do

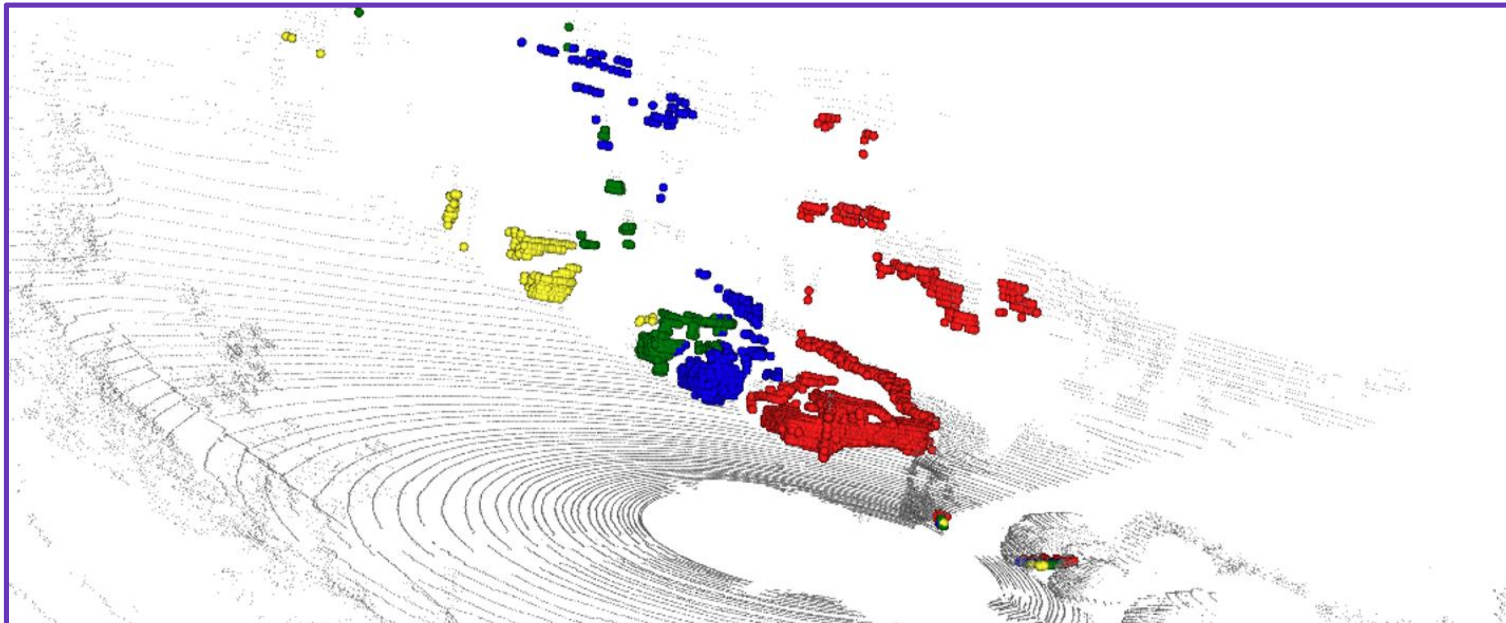


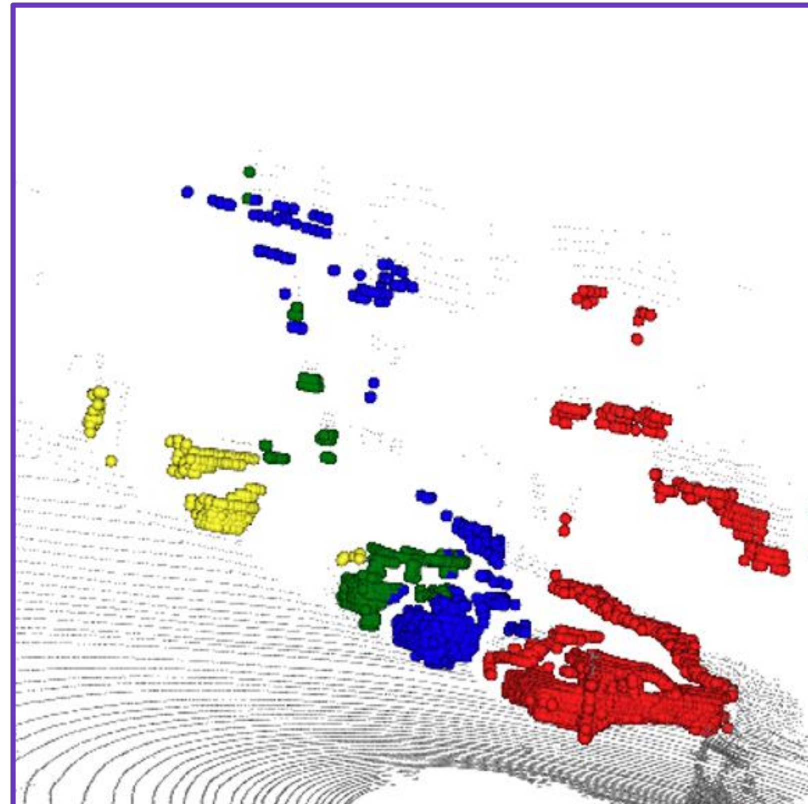
Fig.: The corresponding sub-pointclouds marked in color. We now have a 3D object detector running.

Evaluation of an 3D object detector
What we have to do

So we have a 3D object detector running on our machine. But how good is it?

We can see that we have too many points per object available.

But which points are correctly take as the sub-pointcloud for each detection and which are incorrectly included in the pointcloud?

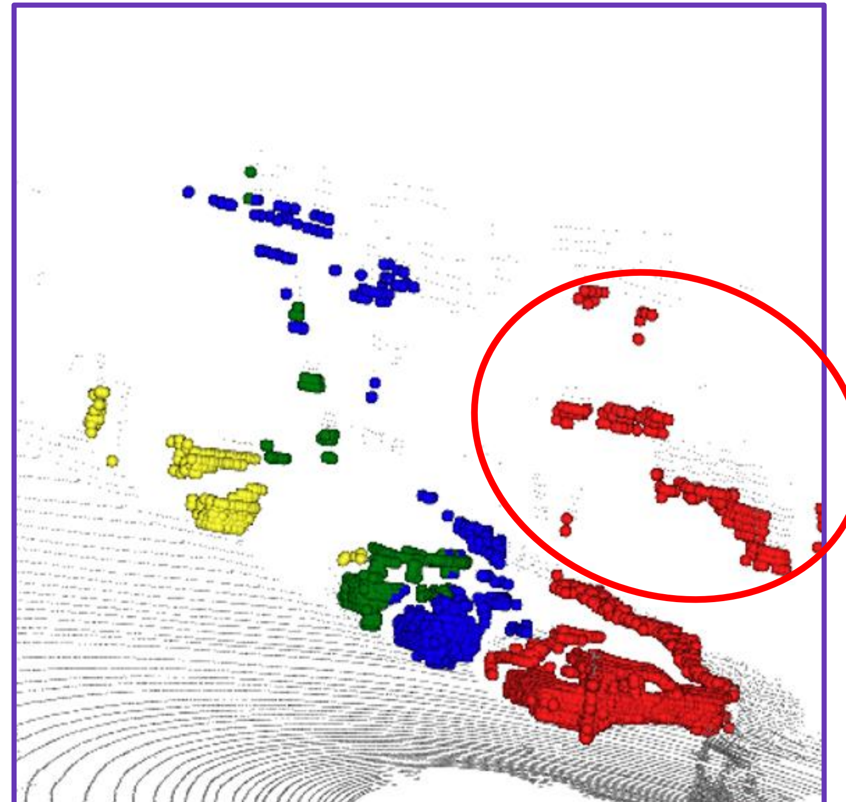


Evaluation of an 3D object detector
What we have to do

You can see this best on the object marked in red.

Here, we have a lot of incorrect points marked with the red circle.

We now want to evaluate how good or bad our detector actually worked.

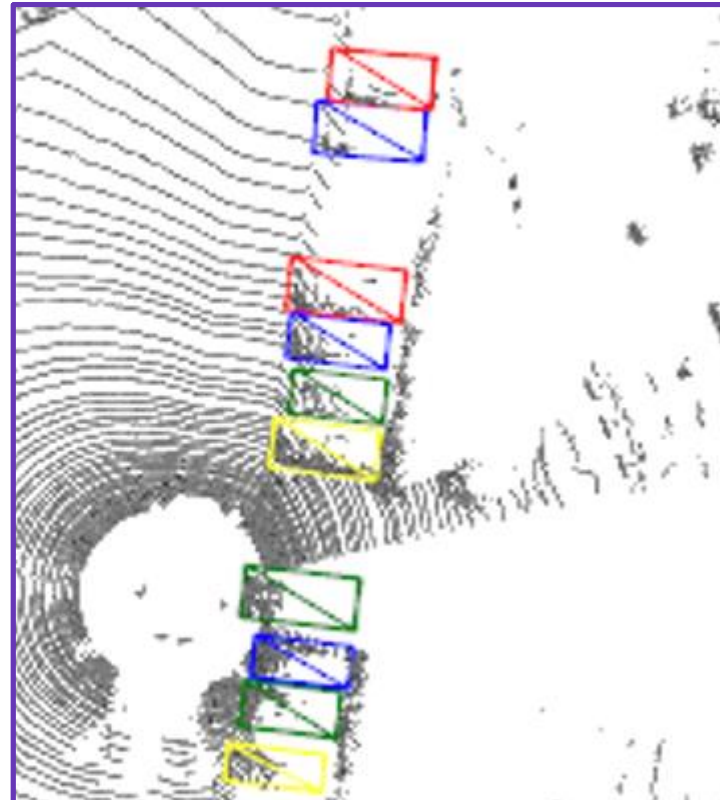


Evaluation of an 3D object detector
What we have to do

For this step, we will use the 3D bounding boxes calculated from the labelled ground truth.

Only the points inside the corresponding bounding box can be considered to be classified correctly as belonging to the corresponding detected objec.

All point outside are wrong.

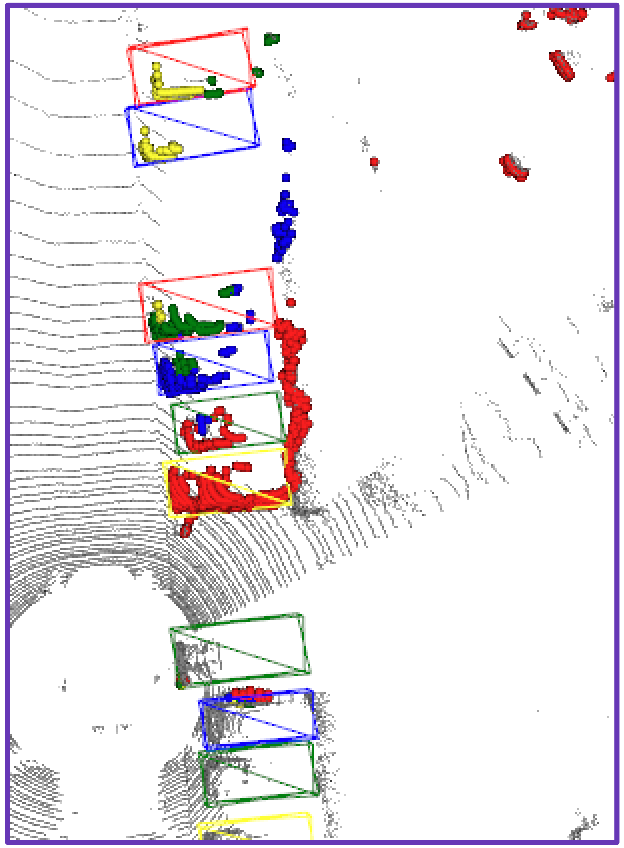


Evaluation of an 3D object detector
What we have to do

For this step, we will use the 3D bounding boxes calculated from the labelled ground truth.

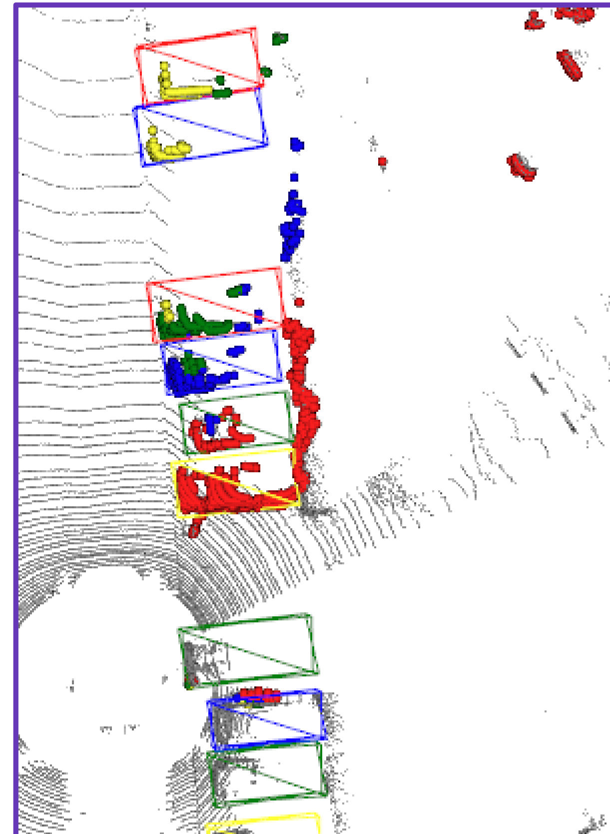
Only the points inside the corresponding bounding box can be considered to be classified correctly as belonging to the corresponding detected objec.

All point outside are wrong.



Evaluation of an 3D object detector
What we have to do

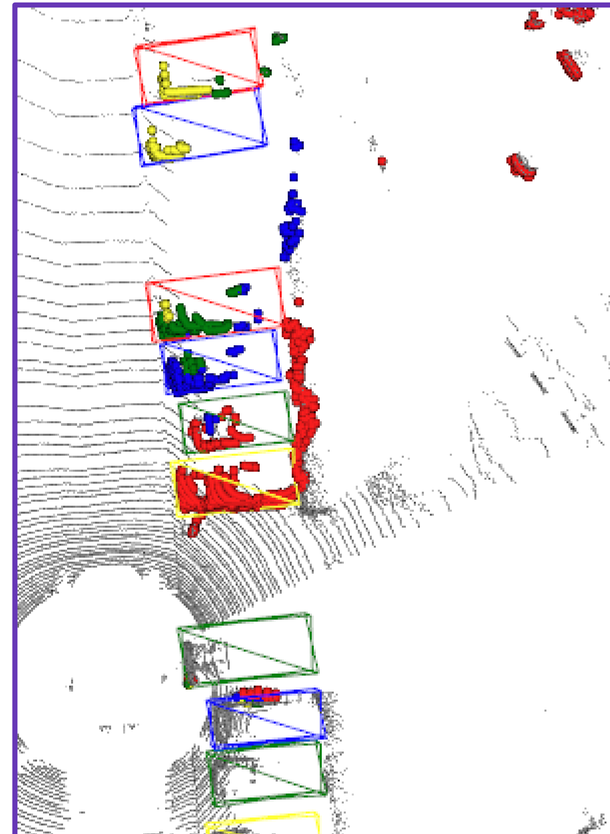
For this task, we will not evaluate the performance of the YOLO sematic instance segmentation but focus on the evaluation of the overall 3D object detector.



Evaluation of an 3D object detector
What we have to do

To be able to evaluate your detector, you need to be able to answer the following questions:

- So how many points are inside or outside?
- Is the absolute amount of points important or the relative amount?
- Which detected object can then be considered a correct detected object?



Evaluation of an 3D object detector
Goals

Goals of this task:

Evaluate the results of your object detector on the dataset provided via moodle. **Evaluate on the car class only.** Provide the following:

- provide an image with the result of the 3D object detector as colour coded sub-pointclouds of each car detection with the 3D bounding boxes as seen above for each scene
- evaluate your 3D object detector as described above
- ...

Evaluation of an 3D object detector
Goals

- ...
- Now, you have the evaluation and some numbers available. But does your solution actually work? Would you use or sell it? **Provide a short discussion whether or not you would use this version in an actual car.**

Provide all information needed to reproduce your algorithms. Answer the **What?**, **How?**, and **Why?** questions described above. Include images where they are helpful to describe your steps.

WHAT IS NEXT?

What is next?

What is next? Master Thesis!Moving on to a bigger Data Set

As part of a scientific project, you could move now fully to a larger data set like KITTI or the Astyx dataset. On this dataset, you could evaluate other object detectors as well.

The KITT leaderboard could be your starting point to select such a detector:

https://www.cvlibs.net/datasets/kitti/eval_object.php?obj_benchmark=3d

This would / could be your first steps into the direction on applied deep learning on sensor data!

What is next?
Finding the optimal fusion

If you are interested in further working on sensor fusion, the implementation and evaluation of your algorithm would be a very good starting point.

Of course there are different approaches to combine the data. In principle, these approaches can be seen as **three different types**:

- low level fusion
- feature level fusion
- high level fusion

What is next?
Finding the optimal fusion

At the moment, there is a lot of research going on with the goal to find an optimal sensor fusion.

If you are interested in this topic, you could do your own research as part of a scientific project or your master thesis.

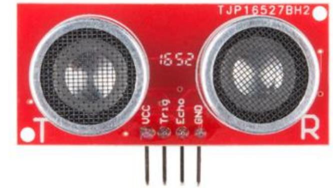
A starting point could be one of the datasets discussed during the lecture. You will find a lot of material how to work with them online…

Your Own Stuff
What did you have in mind?

You always wanted to built a autonomous robot?

We have some robots available as well as several sensors (camera and lidar).

You could use these parts for your project as well!





Your Own Stuff
What did you have in mind?

Also possible would be your own software project.

We can discuss your personal ideas and see if they work as an individual project!

Just send me an email stefan.elser@rwu.de and we can meet to discuss your own project idea!

HOCHSCHULE
RAVENSBURG-WEINGARTEN
UNIVERSITY
OF APPLIED SCIENCES

VIELEN DANK FÜR IHRE AUFMERKSAMKEIT!

Doggenriedstraße
88250 Weingarten

Postfach 3022
88216 Weingarten

www.rwu.de
info@rwu.de