

# Metrics

Let's explore some metrics that will help evaluate the performance of the vision-based self-driving car system.

We'll cover the following ^

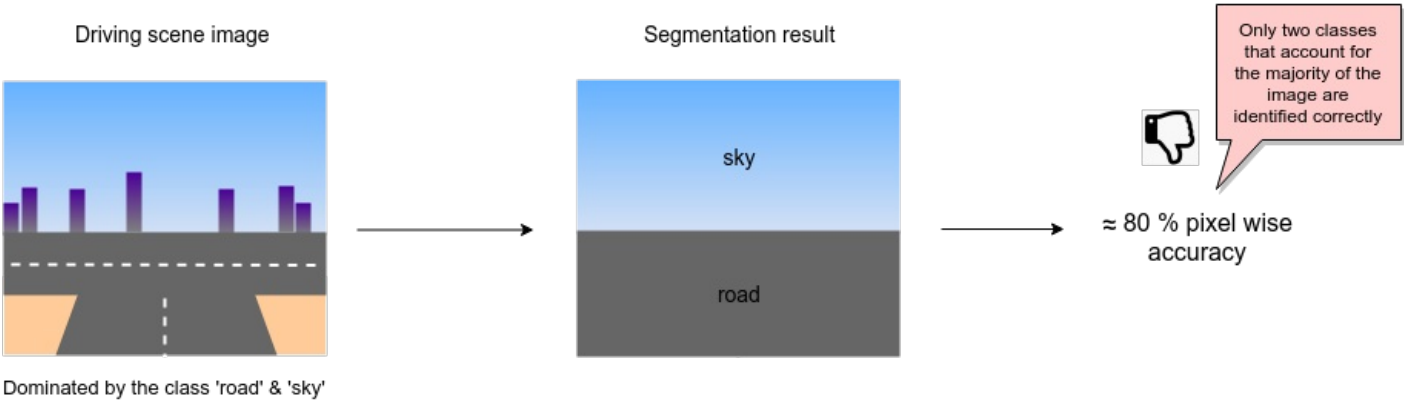
- Component level metric
  - IoU
- End-to-end metric
  - Manual intervention
  - Simulation errors

## Component level metric #

The component of the self-driving car system under discussion here is the semantic segmentation of objects in the input image. In order to look for a suitable metric to measure the performance of an image segmenter, the first notion that comes to mind is the *pixel-wise accuracy*. Using this metric, you can simply compare the ground truth segmentation with the model's predictive segmentation at a pixel level. However, this might not be the best idea, e.g., consider a scenario where the driving scene image has a major class imbalance, i.e., it mostly consists of sky and road.

For this example, assume one-hundred pixels (ground truth) in the driving scene input image and the annotated distribution of these pixels by a human expert is as follows: sky=45, road=35, building=10, roadside=10.

If your model correctly classifies all the pixels of only sky and road (i.e., sky=50, road=50), it will result in high pixel-wise accuracy. However, this is not really indicative of good performance since the segmenter completely misses other classes such as building and roadside!



High pixel-wise accuracy doesn't always translate to a great performance by the segmentation model

In general, you need a higher pixel-wise accuracy for objects belonging to each class as an output from the segmenter. The following metric caters to this requirement nicely.

## IoU #

Intersection over Union (IoU) divides the overlapping area between the predicted segmentation and the ground truth in perspective, by the area of union between the predicted segmentation and the ground truth.

$$IoU = \frac{\text{area of overlap}}{\text{area of union}} \text{ or } IoU = \frac{|P_{pred} \cap P_{gt}|}{|P_{pred} \cup P_{gt}|}$$

This metric ranges from 0–1 (or 0%–100%). ‘0’ indicates no overlap while ‘1’ indicates perfectly overlapping segmentation.

The driving images contain objects of multiple classes, as shown above (e.g., building, roadside, sky, road, etc.). So, you will be performing multi-class segmentation, for which the **mean IoU** is calculated by taking the *average of the IoU for each class*.

Now, let’s calculate the mean IoU and see how it differs from pixel accuracy. You will be considering the same driving scene image and its predicted segmentation, as shown above.

You will begin by calculating the IoU for each class. Here, the “area of overlap” means the number of pixels that belong to the particular class in both the prediction and ground-truth. Whereas, the “area of union” refers to the number of pixels that belong to the particular class in the prediction and in ground-truth, but not in both (the overlap is subtracted).

Let’s apply the calculations: ground truth: [sky=45, road=35, building=10, roadside=10], segmenter predictions: [sky=50, road=50, building=0, roadside=0].

$$IoU_C = \frac{P_{pred} \cap P_{gt}}{(P_{pred} + P_{gt}) - (P_{pred} \cap P_{gt})}$$

where  $P_{pred}$  = number of pixels classified as class  $C$  in prediction and  $P_{gt}$  = number of pixels classified as class  $C$  in ground truth

$$IoU_{sky} = \frac{45}{(50+45)-45} = 90\%$$

$$IoU_{road} = \frac{35}{(50+35)-35} = 70\%$$

$$IoU_{building} = \frac{0}{(0+10)-0} = 0\%$$

$$IoU_{road\ side} = \frac{0}{(0+10)-0} = 0\%$$

$$Mean\ IoU = \frac{IoU_{road} + IoU_{sky} + IoU_{road\ side} + IoU_{building}}{4} = \frac{90+70+0+0}{4} = 40\%$$

You can see that Mean IoU (40%) is significantly lower than the pixel-wise accuracy (80%) and provides an accurate picture of how well your segmentation is performing.

You will be using IoU as an offline metric to test the performance of the segmentation model.

## End-to-end metric #

You also require an online, end-to-end metric to test the overall performance of the self-driving car system as you plug in your new image segmenter to see its effect.

## Manual intervention #

Ideally, you want the system to be as close to self-driving as possible, where the person never has to intervene and take control of the driving. So, you can use manual intervention as a metric to judge the success of the overall system. If a person rarely has to intervene, it means that your system is performing well.

This is a good metric for the early testing phase of the self-driving car system. During this time you have a person ready to take over in case the self-driving system makes a poor decision.

## Simulation errors #

Another approach is to use historical data, such as driving scene recording, where an expert driver was driving the car. You will give the historical data as input to your self-driving car system with the new segmentation model and see how its decisions align with the decisions made by an expert driver.

You will assume that the decisions made by the professional driver in that actual scenario are your ground truths. The overall objective will be to minimize the movement and planning errors with these ground truths.

As you replay the data with new segmenter, you will experiment to measure whether your segmentation is resulting in a reduction of your simulation-based errors or not. This will be a good end to end metric to track.

[← Back](#)

Problem Statement

[Next →](#)

Architectural Components

☒ Mark as Completed

 [Report an Issue](#)  [Ask a Question](#)  
([https://discuss.educative.io/tag/metrics\\_\\_self-driving-car-image-segmentation\\_\\_grokking-the-machine-learning-interview](https://discuss.educative.io/tag/metrics__self-driving-car-image-segmentation__grokking-the-machine-learning-interview))