

Vorabfragen

- Ihr CPS soll in einer Fabrik Werkzeuge erkennen, darunter eine selten verwendete Spezialzange. Sie laden ein fertig trainiertes YOLO-Modell herunter. Was passiert vermutlich?
 - a) Es erkennt die Zange, weil sie ein Objekt ist.
 - b) Es erkennt sie nur, wenn sie einer der trainierten Kategorien ähnelt, und kann sie dann falsch benennen.
 - c) Es meldet “unbekanntes Objekt”.
 - d) Es lernt die Zange automatisch beim ersten Sehen.
- Sie trainieren einen Roboter in einer Simulation. Je realistischer die Simulation wird, desto besser sollte er in der Realität funktionieren. Stimmt das? Nennen Sie einen Grund dafür oder dagegen.
- Sie stehen mit verbundenen Augen in einem unbekannten Raum und sollen eine Karte zeichnen. Sie dürfen nur zählen, wie viele Schritte Sie in welche Richtung gehen. Was wird schiefgehen, und was bräuchten Sie, um den Fehler zu korrigieren?



Cyber-Physische Systeme – WS 26/27

Gelesen von Prof. Dr. Daniel Gaida

13.10.2026

Prof. Dr. Daniel Gaida

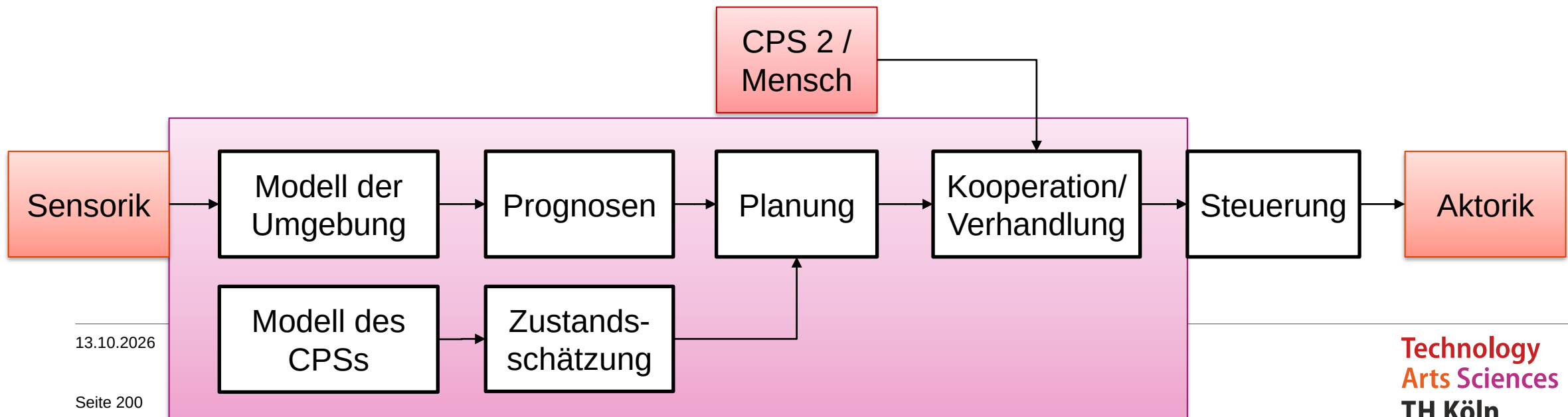
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

Technology
Arts Sciences
TH Köln

Inhalt

- Modell der Umgebung



13.10.2026

Seite 200

Inhalt

Modellierung der Umgebung von CPS:

- Modellierung mit KI/Deep Learning
 - Objektdetektion, -tracking, Bildsegmentierung, Greifpunkte
 - Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?
 - Spracherkennung
 - Verhaltensmodellierung von Fußgängern, Fahrzeugen
 - World Models
- -
 -
- -
 -

Inhalt

Modellierung der Umgebung von CPS:

- Modellierung mit KI/Deep Learning
 - Objektdetektion, -tracking, Bildsegmentierung, Greifpunkte
 - Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?
 - Spracherkennung
 - Verhaltensmodellierung von Fußgängern, Fahrzeugen
 - World Models
- Lernen in virtuellen Umgebungen (in einer Simulation)
 - 3D Simulation
 - Von der Simulation in die Realität (Sim2real)
- -
 -

Inhalt

Modellierung der Umgebung von CPS:

- Modellierung mit KI/Deep Learning
 - Objektdetektion, -tracking, Bildsegmentierung, Greifpunkte
 - Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?
 - Spracherkennung
 - Verhaltensmodellierung von Fußgängern, Fahrzeugen
 - World Models
- Lernen in virtuellen Umgebungen (in einer Simulation)
 - 3D Simulation
 - Von der Simulation in die Realität (Sim2real)
- Statische Modelle der Umgebung (Karte)
 - SLAM – Simultaneous Localization and Mapping
 - Semantische Karte (Karte mit Objekten/Orten: Couch, TV, Küche, ...)

Lernziel

Die Studierenden können geeignete Technologien zur Modellierung der Umgebung eines CPS auswählen, indem sie

- die Umgebung ihres CPS identifizieren und beschreiben,
- aus Deep Learning basierten Technologien wie Objekt Detektion, Objekt Tracking, Bildsegmentierung und Spracherkennung geeignete Technologien wählen,
- im Bedarfsfall 3D Simulationen zur Erstellung von Trainingsdaten nutzen,
- eine Karte der Umgebung erstellen und
- geeignete Bibliotheken und Tools zur Modellierung der Umgebung des CPS wählen,

um später die Umgebung von Cyber-physischen Systemen modellieren zu können.

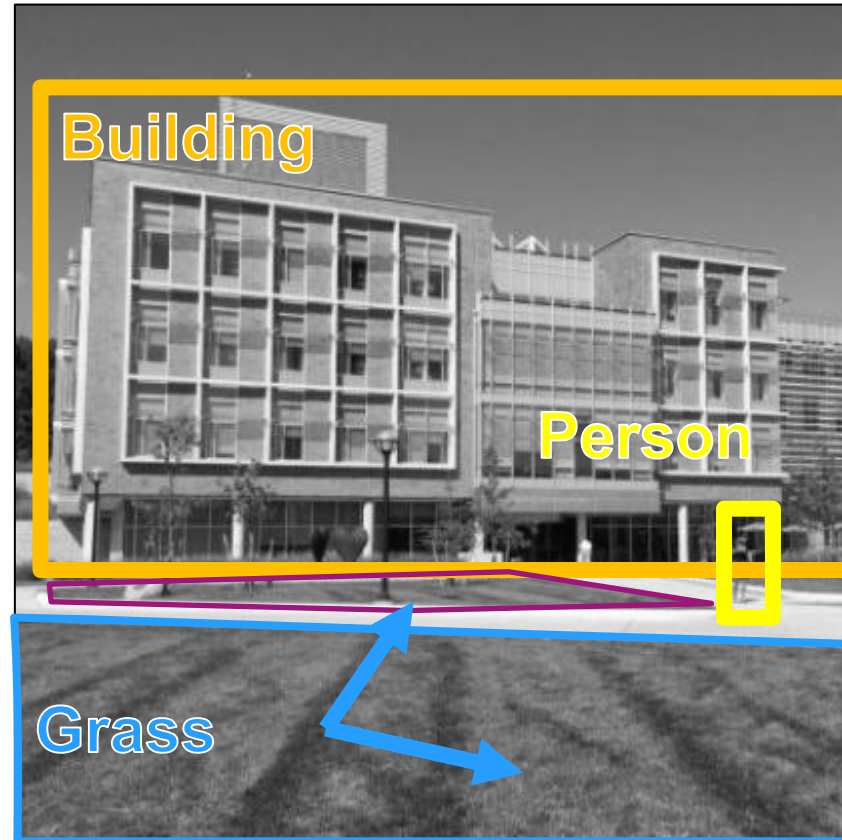
Welche Rolle spielt Computer Vision für CPS?

Ziel: CPS muss mit Umgebung interagieren können



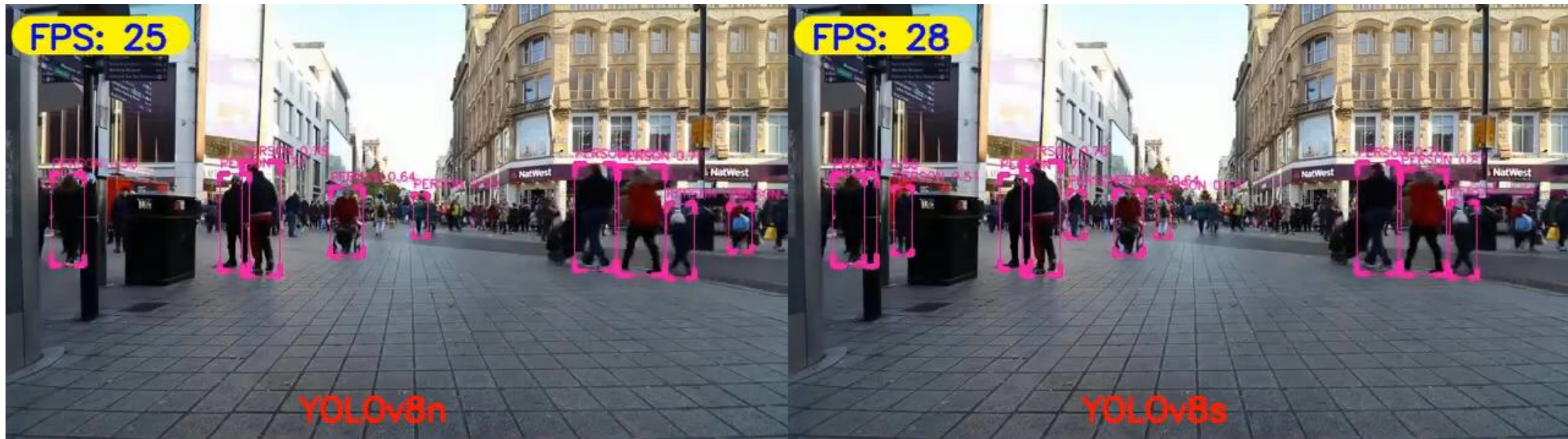
Welche Rolle spielt Computer Vision für CPS?

Ziel: Benennung der Objekte in der Szene - Objektdetektion



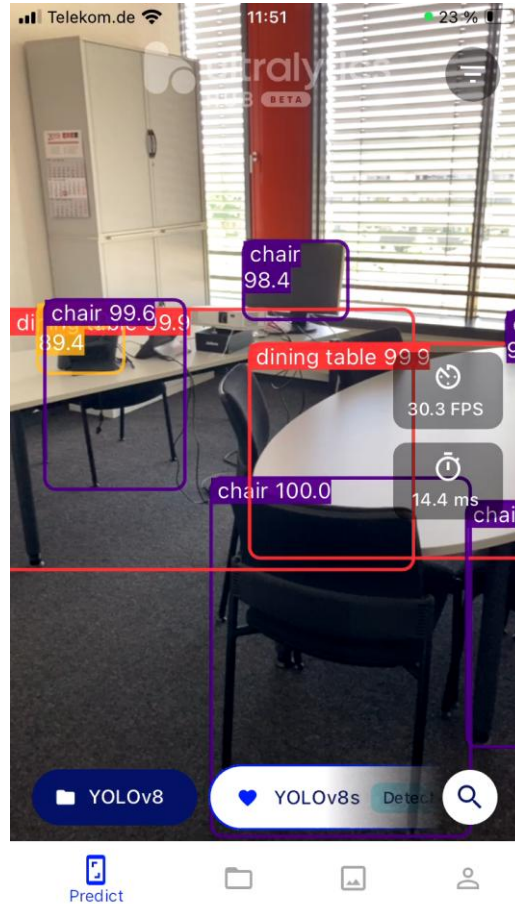
Welche Rolle spielt Computer Vision für CPS?

Objektdetektion



Welche Rolle spielt Computer Vision für CPS?

Objektdetektion



- YOLOv8
- Erkennt nur 80 verschiedene Objektkategorien (Autos, Fahrräder, Tiere, Regenschirme, ...)



Welche Rolle spielt Computer Vision für CPS?

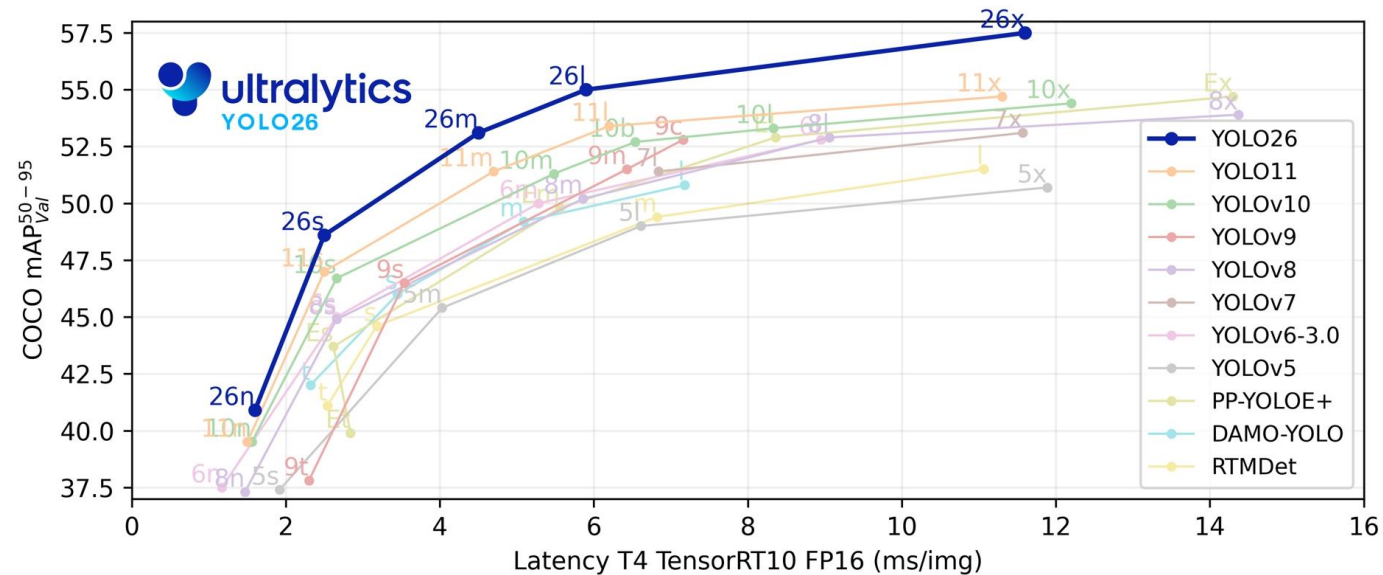
Objektdetektion

Deep Learning Modell

- YOLO – You only look once
- Redmon, J., & Farhadi, A. (2015). You only look once: Unified, real-time object detection. In Proc. IEEE CVPR (pp. 779-788).

- YOLO26 – plug & play

<https://github.com/ultralytics/ultralytics>



Welche Rolle spielt Computer Vision für CPS?

Objektdetektion

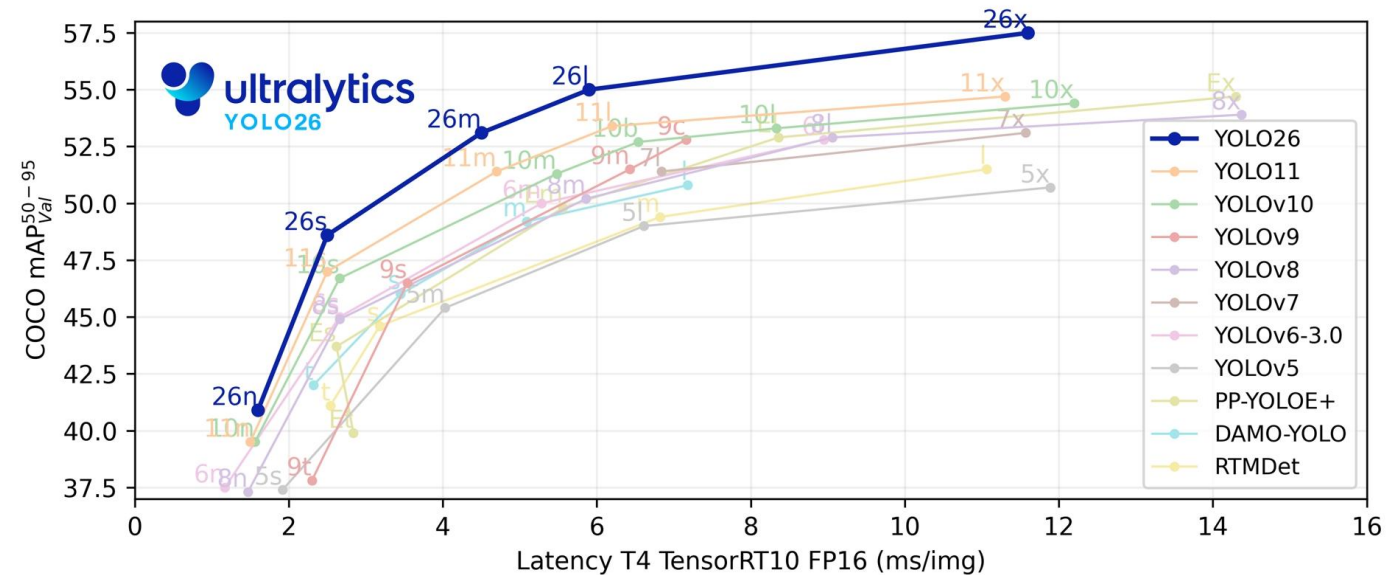
Deep Learning Modell

- YOLO – You only look once
- Redmon, J., & Farhadi, A. (2015). You only look once: Unified, real-time object detection. In Proc. IEEE CVPR (pp. 779-788).

- YOLO26 – plug & play

<https://github.com/ultralytics/ultralytics>

- Fertig trainierte Modelle in unterschiedlichen Größen herunterladbar

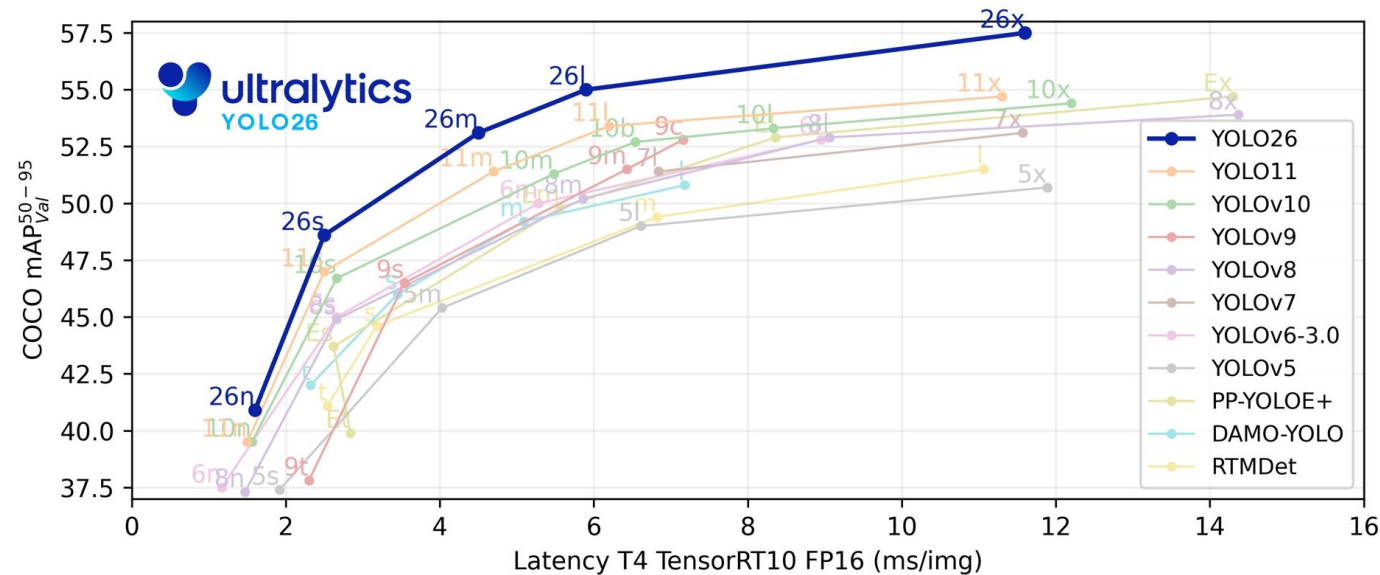


Welche Rolle spielt Computer Vision für CPS?

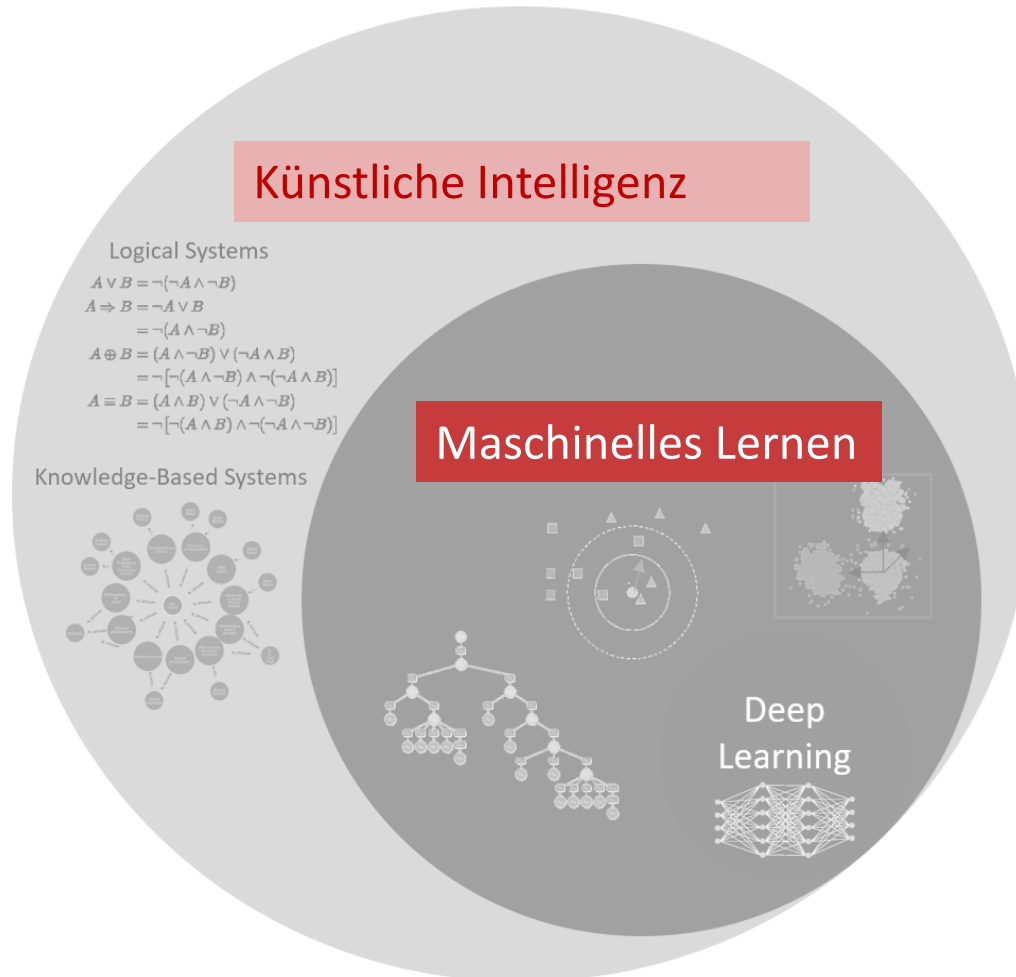
Objektdetektion

Deep Learning Modell

- YOLO – You only look once
 - Redmon, J., & Farhadi, A. (2015). You only look once: Unified, real-time object detection. In Proc. IEEE CVPR (pp. 779-788).
- YOLO26 – plug & play
 - <https://github.com/ultralytics/ultralytics>
- Fertig trainierte Modelle in unterschiedlichen Größen herunterladbar
- Trainiert an COCO Datensatz
 - 80 verschiedene Objektkategorien (Autos, Fahrräder, Tiere, Regenschirme, ...)



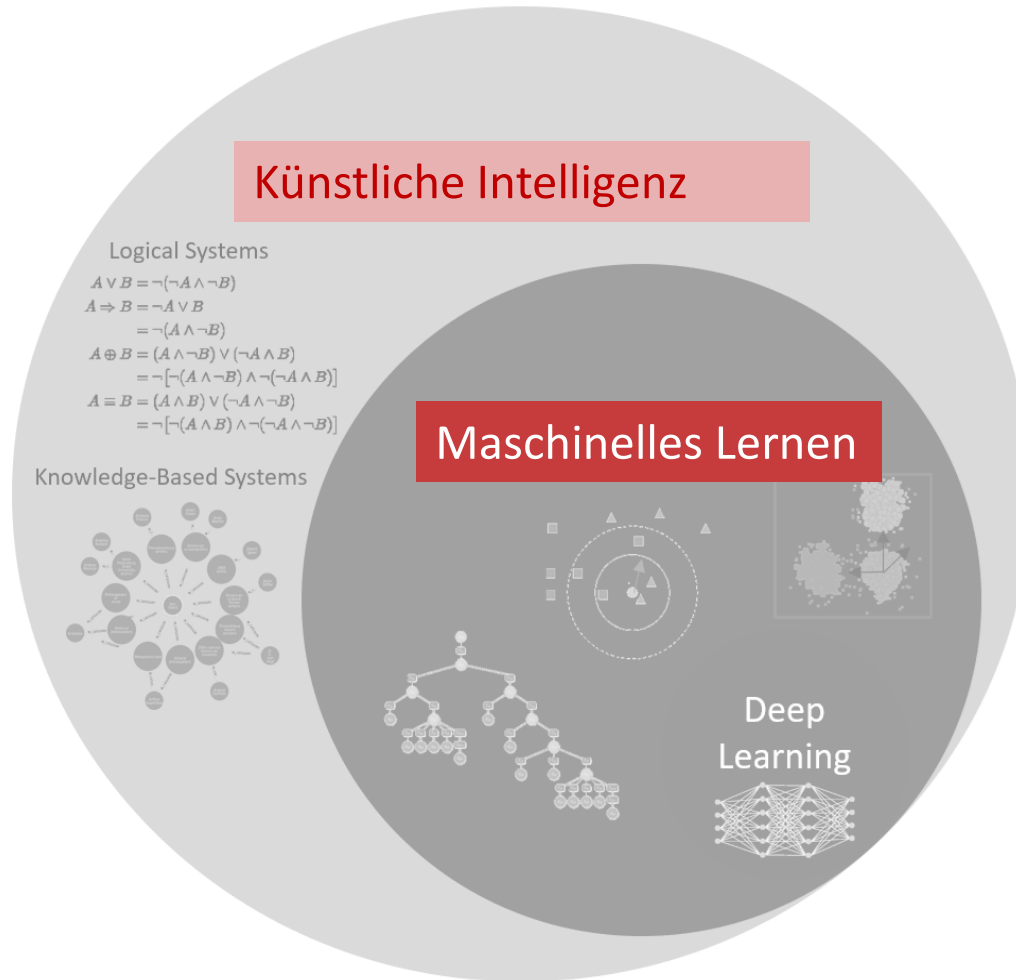
KI, Machine Learning und Deep Learning – Was ist der Unterschied?



- Künstliche Intelligenz als Oberbegriff
 - Logisches Schließen, wissensbasierte Systeme
 - Optimierung
 - Statistische Analyse (maschinelles Lernen)

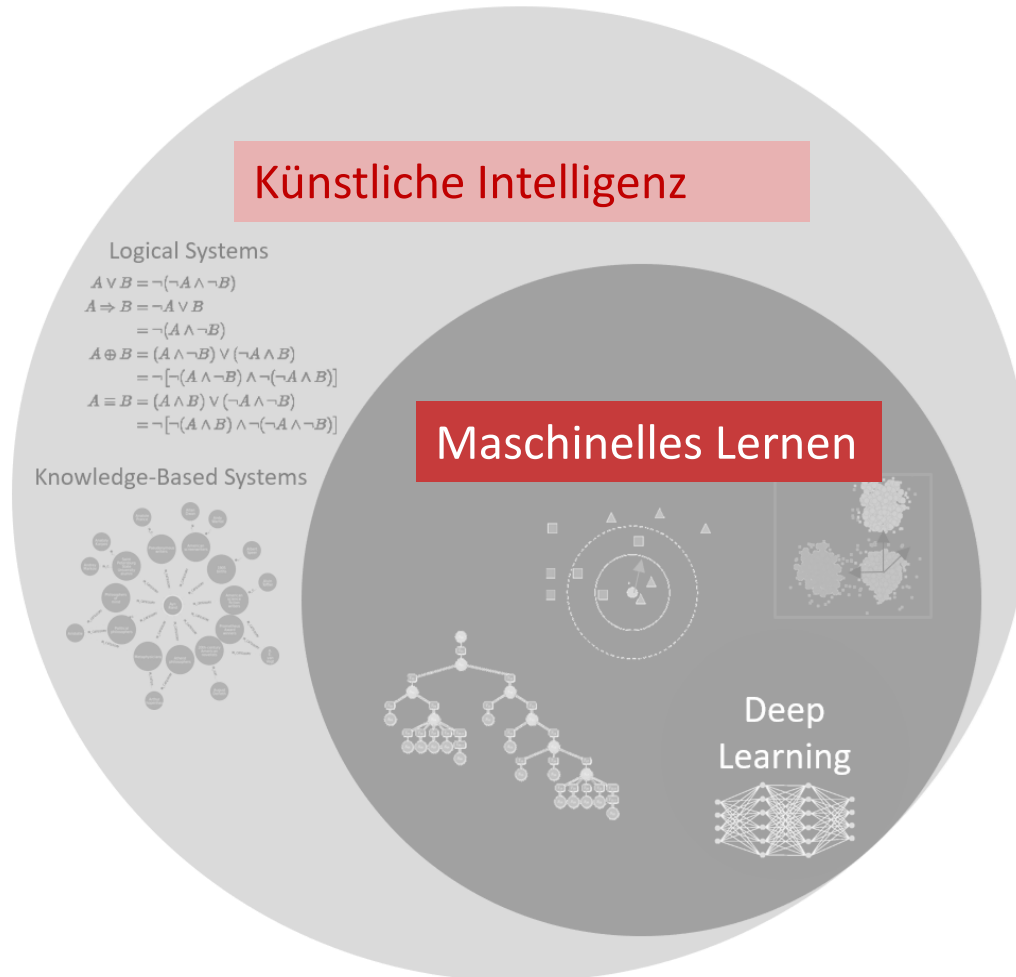
Künstliche Intelligenz (KI) beschäftigt sich mit Methoden, die es einem Computer ermöglichen, solche Aufgaben zu lösen, die, wenn sie vom Menschen gelöst werden, Intelligenz erfordern.

KI, Machine Learning und Deep Learning – Was ist der Unterschied?



- Künstliche Intelligenz als Oberbegriff
 - Logisches Schließen, wissensbasierte Systeme
 - Optimierung
 - Statistische Analyse (maschinelles Lernen)
- Maschinelles Lernen
 - Systeme die aus Erfahrung & Beispielen lernen
 - Erkennen von Mustern in Daten
 - Eigenständiges Erlernen von Regeln

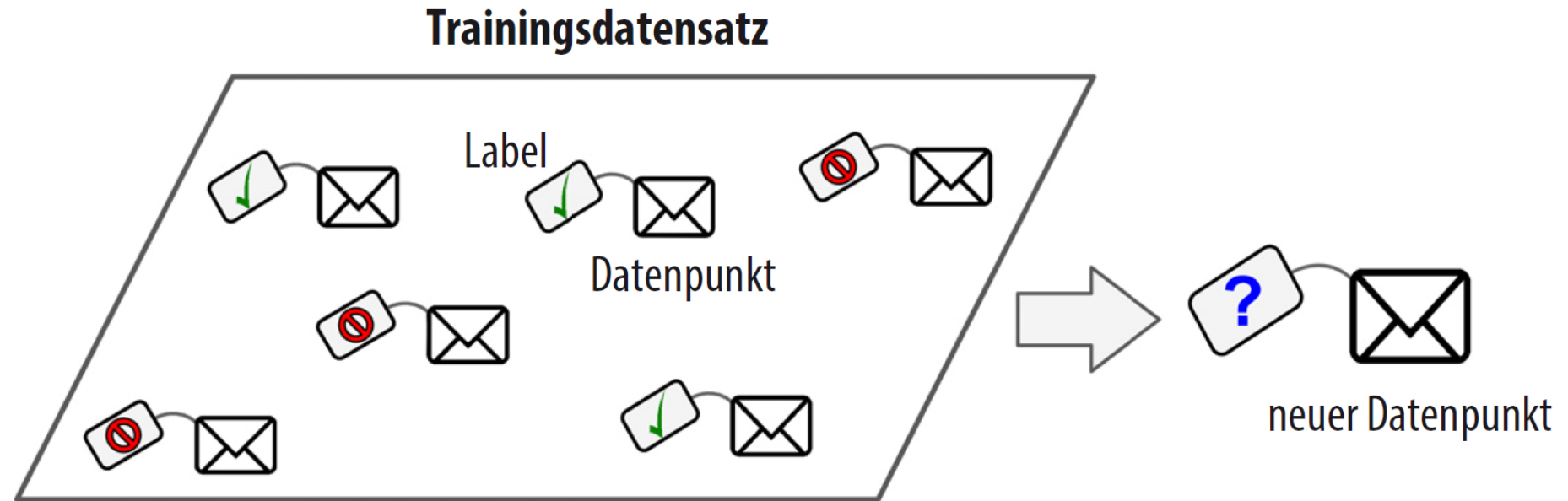
KI, Machine Learning und Deep Learning – Was ist der Unterschied?



- Künstliche Intelligenz als Oberbegriff
 - Logisches Schließen, wissensbasierte Systeme
 - Optimierung
 - Statistische Analyse (maschinelles Lernen)
- Maschinelles Lernen
 - Systeme die aus Erfahrung & Beispielen lernen
 - Erkennen von Mustern in Daten
 - Eigenständiges Erlernen von Regeln
- Deep Learning
 - Ein Teilgebiet des maschinellen Lernens
 - Hype durch Erfolge in Bild- und Spracherkennung und Sprachverarbeitung

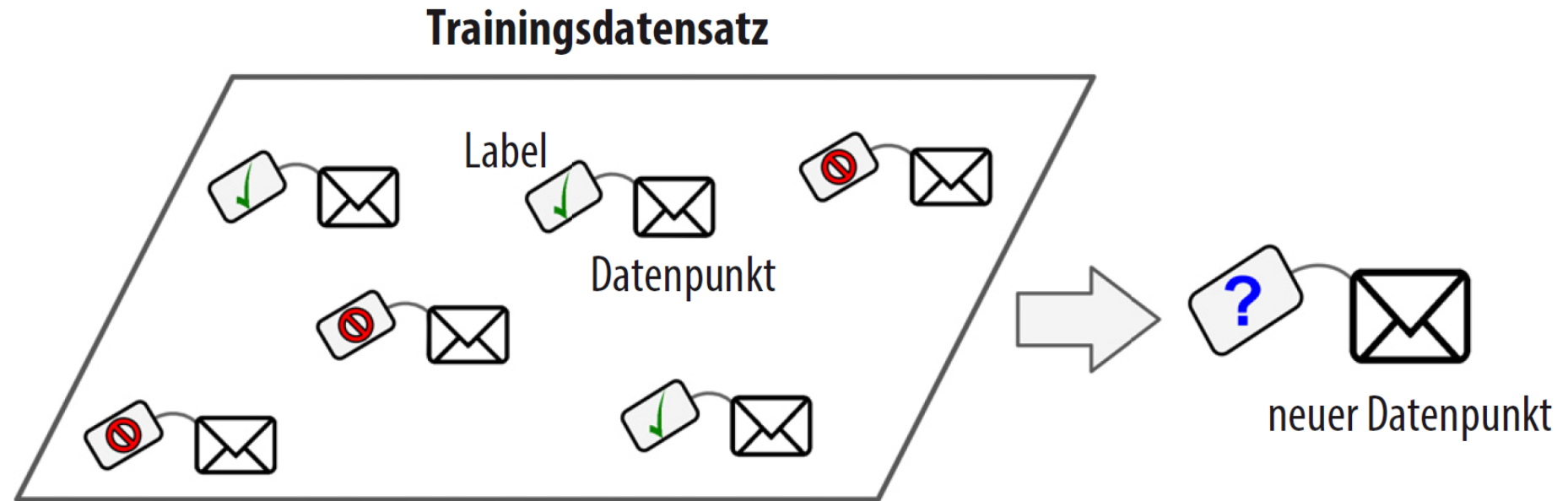
Was ist Maschinelles Lernen?

- Ein Spamfilter ist ein maschinelles Lernprogramm, das aus Beispielen für Spam-E-Mails (z.B. vom Nutzer gelabelten) und gewöhnlichen E-Mails (Nicht-Spam) lernt, Spam zu erkennen.
-



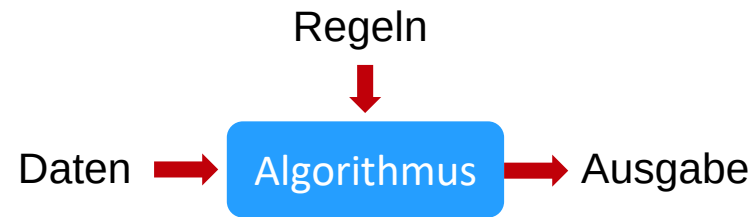
Was ist Maschinelles Lernen?

- Ein Spamfilter ist ein maschinelles Lernprogramm, das aus Beispielen für Spam-E-Mails (z.B. vom Nutzer gelabelten) und gewöhnlichen E-Mails (Nicht-Spam) lernt, Spam zu erkennen.
- Maschinelles Lernen lernt diese Aufgabe aus dem Trainingsdatensatz, in dem es Muster/Zusammenhänge erkennt und wendet das erlernte Wissen auf neue E-Mails an, um Spam zu erkennen.



Warum wird Maschinelles Lernen verwendet?

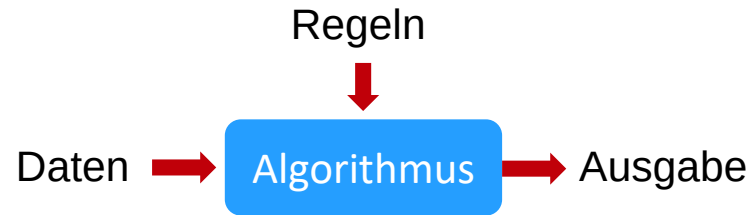
Traditionelles Programmieren vs. Maschinelles Lernen



Traditionelle Programmierung

Warum wird Maschinelles Lernen verwendet?

Traditionelles Programmieren vs. Maschinelles Lernen

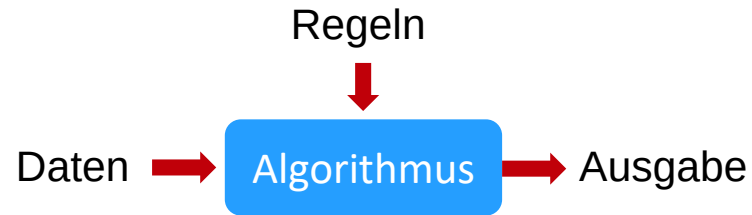


Traditionelle Programmierung

- Alle Regeln werden explizit programmiert
- Jede Regel basiert auf einer logischen Grundlage
- Der Algorithmus gibt eine Ausgabe aus, die der logischen Anweisung folgt
- Komplexe Systeme benötigen viele Regeln

Warum wird Maschinelles Lernen verwendet?

Traditionelles Programmieren vs. Maschinelles Lernen



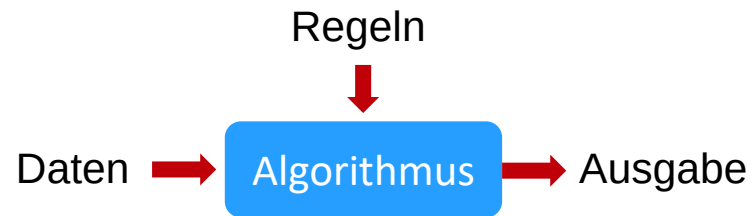
Traditionelle Programmierung

- Alle Regeln werden explizit programmiert
- Jede Regel basiert auf einer logischen Grundlage
- Der Algorithmus gibt eine Ausgabe aus, die der logischen Anweisung folgt
- Komplexe Systeme benötigen viele Regeln

→ Instandhaltung kann schnell untragbar werden

Warum wird Maschinelles Lernen verwendet?

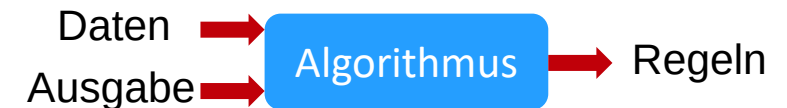
Traditionelles Programmieren vs. Maschinelles Lernen



Traditionelle Programmierung

- Alle Regeln werden explizit programmiert
- Jede Regel basiert auf einer logischen Grundlage
- Der Algorithmus gibt eine Ausgabe aus, die der logischen Anweisung folgt
- Komplexe Systeme benötigen viele Regeln

→ Instandhaltung kann schnell untragbar werden

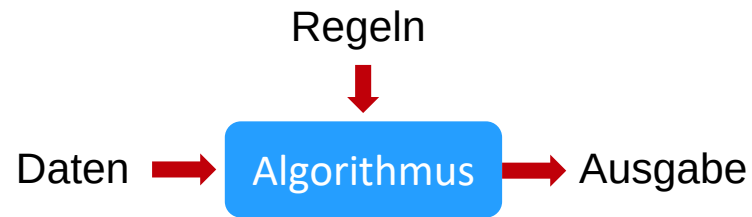


Maschinelles Lernen

- Soll dieses Problem überwinden

Warum wird Maschinelles Lernen verwendet?

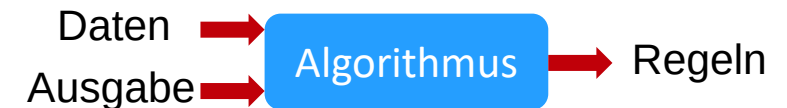
Traditionelles Programmieren vs. Maschinelles Lernen



Traditionelle Programmierung

- Alle Regeln werden explizit programmiert
- Jede Regel basiert auf einer logischen Grundlage
- Der Algorithmus gibt eine Ausgabe aus, die der logischen Anweisung folgt
- Komplexe Systeme benötigen viele Regeln

→ Instandhaltung kann schnell untragbar werden



Maschinelles Lernen

- Soll dieses Problem überwinden
- Regeln werden basierend auf der Korrelation zwischen Daten und Ausgabe erlernt
- Regeln müssen nicht explizit geschrieben werden
- Wenn sich Daten ändern, können Regeln aktualisiert werden

Training von Deep Learning Modellen

Objektdetektion

Bevor Sie ein Deep Learning Modell selbst erstellen und trainieren:

- Es gibt sehr viele an sehr großen Datensätzen trainierte Modelle, die Sie herunterladen und direkt nutzen können
- Sie können dieses Modell auch als initiales Modell nutzen, welches Sie mit Ihren eigenen Daten verbessern können
 - Transfer Learning
 - Knowledge Distillation

■

■

■

Training von Deep Learning Modellen

Objektdetektion

Bevor Sie ein Deep Learning Modell selbst erstellen und trainieren:

- Es gibt sehr viele an sehr großen Datensätzen trainierte Modelle, die Sie herunterladen und direkt nutzen können
- Sie können dieses Modell auch als initiales Modell nutzen, welches Sie mit Ihren eigenen Daten verbessern können
 - Transfer Learning
 - Knowledge Distillation
- YOLO26 als Beispiel
- <https://huggingface.co/models>
- <https://modelzoo.co/>

Objektdetektion mit Deep Learning

Trainingsdatensatz

Trainingsdaten

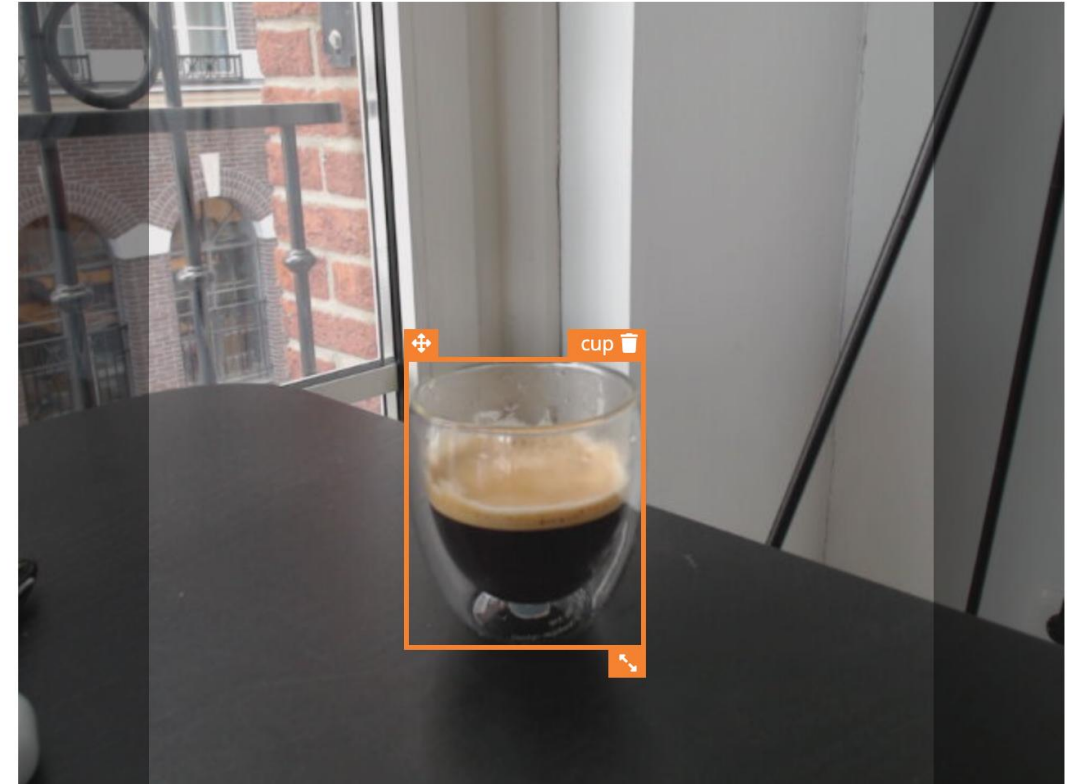
- Bilder

Markieren der Bilder (Labeling)

- Um jedes zu detektierende Objekt ist ein Rahmen zu zeichnen
- Pro Bild können mehrere Objekte markiert werden

-
-
-

Use your mouse to drag a box around an object to add a label. Then click **Save labels** to advance to the next item.



<https://www.edgeimpulse.com/blog/3-ways-to-do-ai-assisted-labeling-for-object-detection>

Objektdetektion mit Deep Learning

Trainingsdatensatz

Trainingsdaten

- Bilder

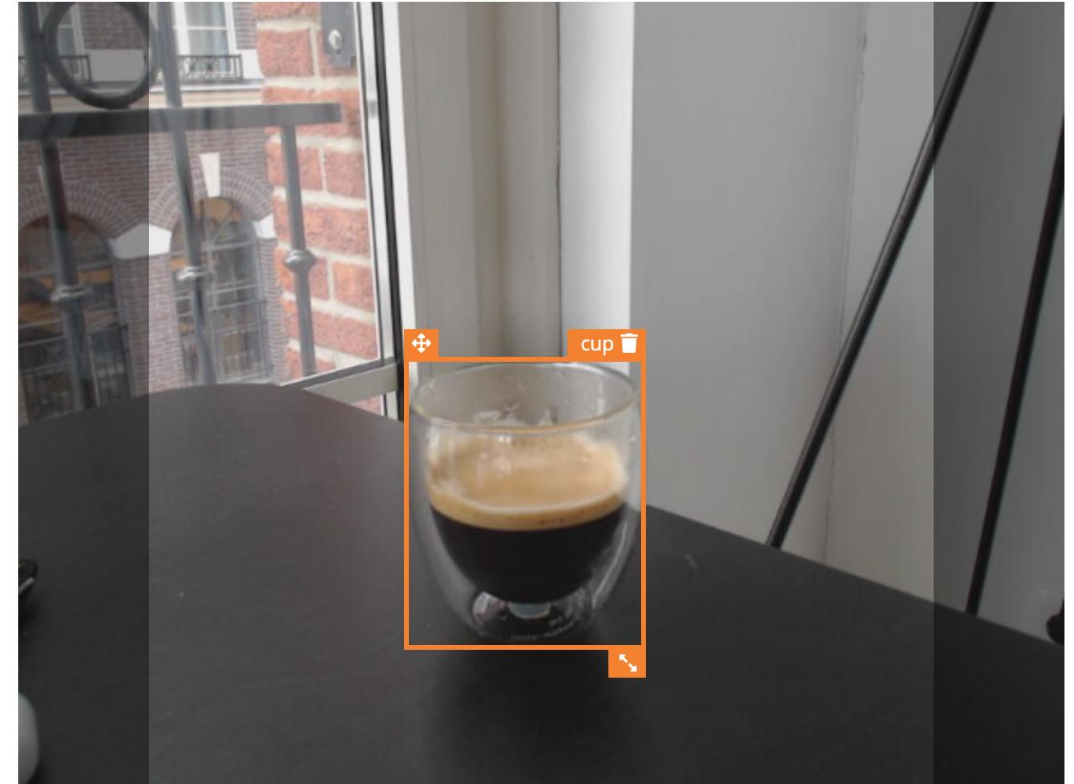
Markieren der Bilder (Labeling)

- Um jedes zu detektierende Objekt ist ein Rahmen zu zeichnen
- Pro Bild können mehrere Objekte markiert werden

Label

- Koordinaten des Rahmens
- Klasse des Objekts
-

Use your mouse to drag a box around an object to add a label. Then click **Save labels** to advance to the next item.



<https://www.edgeimpulse.com/blog/3-ways-to-do-ai-assisted-labeling-for-object-detection>

Objektdetektion mit Deep Learning

Trainingsdatensatz

```
0 0.0414228 0.659696 0.0828456 0.299007
11 0.0199709 0.268372 0.0397828 0.0651804
0 0.349451 0.584807 0.152251 0.418149
0 0.913659 0.58147 0.172332 0.45871
```

Trainingsdaten

- Bilder

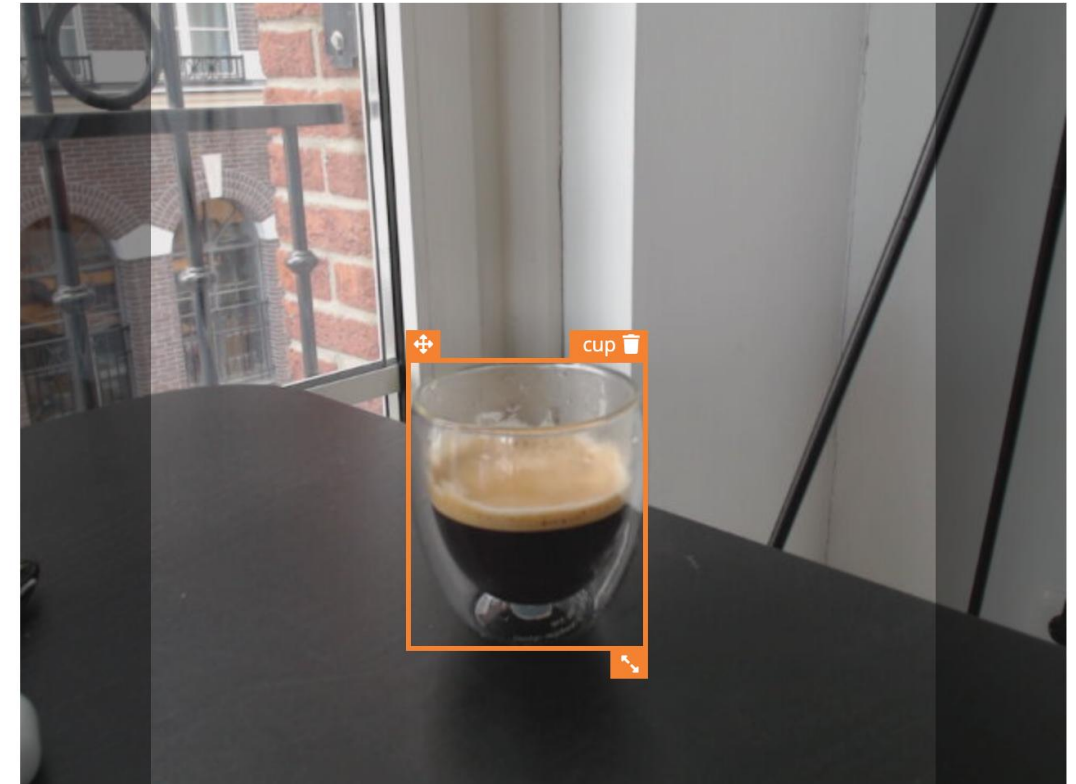
Markieren der Bilder (Labeling)

- Um jedes zu detektierende Objekt ist ein Rahmen zu zeichnen
- Pro Bild können mehrere Objekte markiert werden

Label

- Koordinaten des Rahmens
- Klasse des Objekts
- Für jedes Bild stehen Label in einer txt-Datei

Use your mouse to drag a box around an object to add a label. Then click **Save labels** to advance to the next item.



<https://www.edgeimpulse.com/blog/3-ways-to-do-ai-assisted-labeling-for-object-detection>

Objektdetektion mit Deep Learning

Kostenlose Tools, die das Labeling beschleunigen

- <https://www.cvat.ai/>
- <https://roboflow.com/#annotate>
- <https://pypi.org/project/labelImg/>
- ...

-
-
-
-

Objektdetektion mit Deep Learning

Kostenlose Tools, die das Labeling beschleunigen

- <https://www.cvat.ai/>
- <https://roboflow.com/#annotate>
- <https://pypi.org/project/labelImg/>
- ...

Bereits gelabelte Datensätze (Erster Schritt):

- <https://paperswithcode.com/datasets?task=object-detection&page=1> (233 Object Det. Datensätze)
- Autonomes Fahren, Medizin, ...
- <https://datasetsearch.research.google.com/>
-

Objektdetektion mit Deep Learning

Kostenlose Tools, die das Labeling beschleunigen

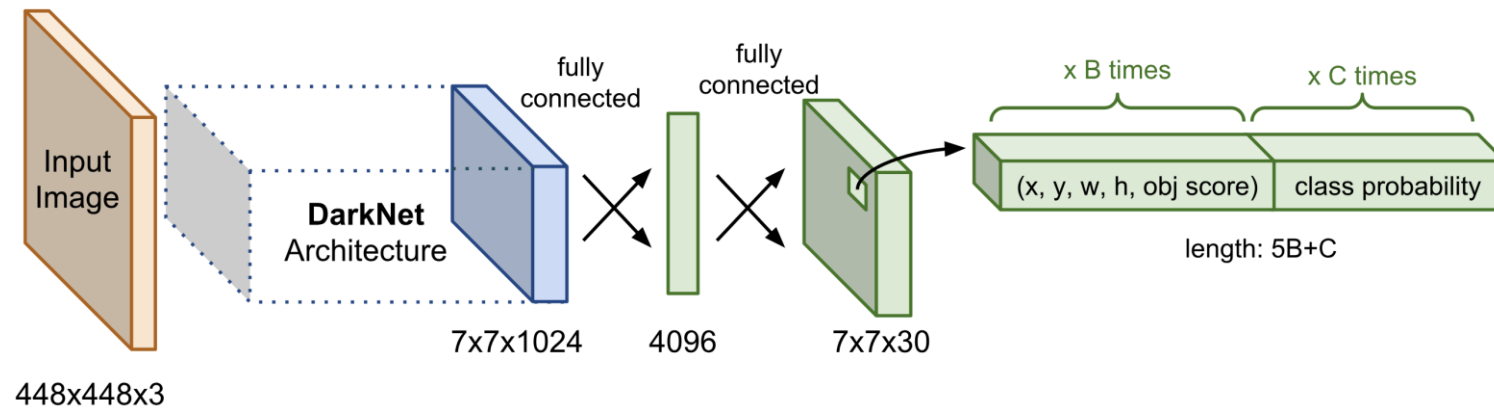
- <https://www.cvat.ai/>
- <https://roboflow.com/#annotate>
- <https://pypi.org/project/labelImg/>
- ...

Bereits gelabelte Datensätze (Erster Schritt):

- <https://paperswithcode.com/datasets?task=object-detection&page=1> (233 Object Det. Datensätze)
- Autonomes Fahren, Medizin, ...
- <https://datasetsearch.research.google.com/>
- Wenn nicht genau der richtige Datensatz dabei ist, hilft es meistens dennoch das Modell auf dem ähnlichsten Datensatz zu trainieren und das Modell dann auf einem eigenen Datensatz weiter zu trainieren (Transfer Learning, finetuning)

Objektdetektion mit Deep Learning

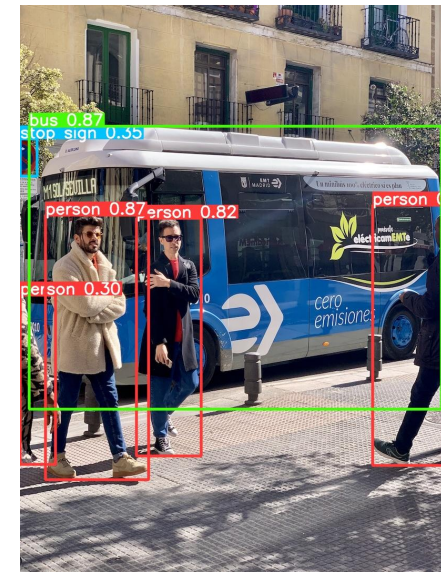
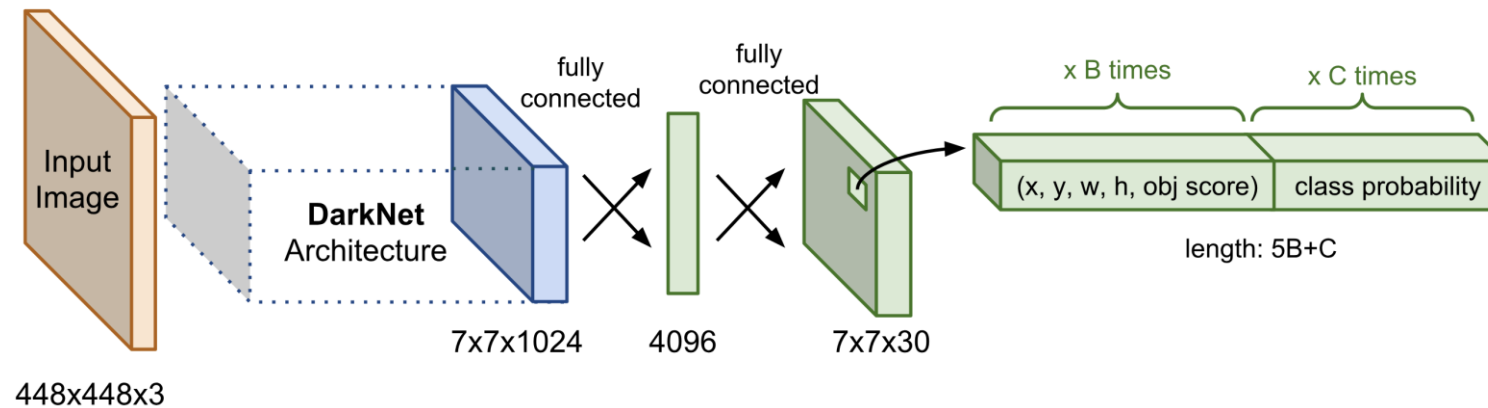
Training – sehr viele Bilder und deren Label



```
0 0.0414228 0.659696 0.0828456 0.299007
11 0.0199709 0.268372 0.0397828 0.0651804
0 0.349451 0.584807 0.152251 0.418149
0 0.913659 0.58147 0.172332 0.45871
```

Objektdetektion mit Deep Learning

Inferenz



YOLO Objektdetektion testen

`pip install ultralytics`

`from ultralytics import YOLO`

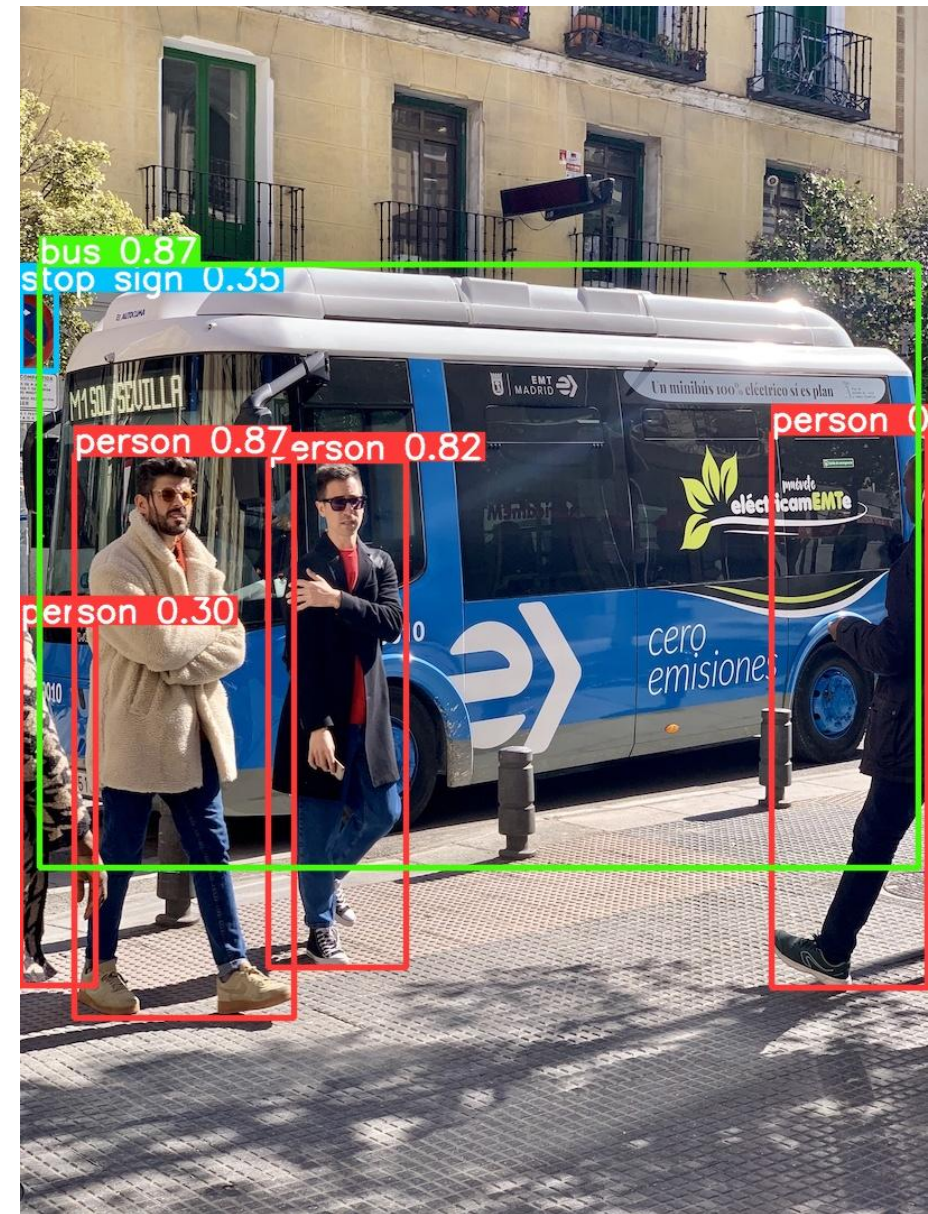
`model = YOLO("yolo26n.pt") # load an official model`

`# Predict with the model`

`results = model("https://ultralytics.com/images/bus.jpg")`

<https://docs.ultralytics.com/tasks/detect/>

Siehe Notebook: [01_objektdetektion_yolo26.ipynb](#)

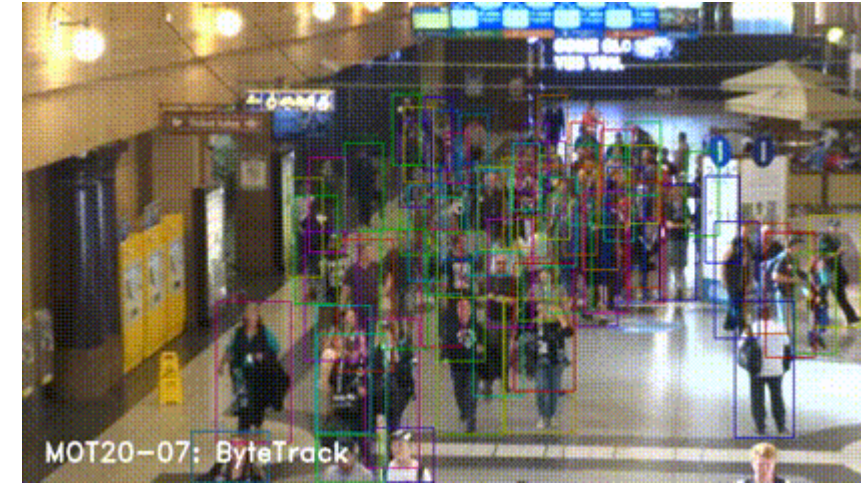


Welche Rolle spielt Computer Vision für CPS?

Ziel: Objekt Tracking

Objekt Tracking

- Eindeutige Objekt Detektion von beweglichen Objekten über eine Sequenz von Frames



<https://github.com/ifzhang/ByteTrack>

Welche Rolle spielt Computer Vision für CPS?

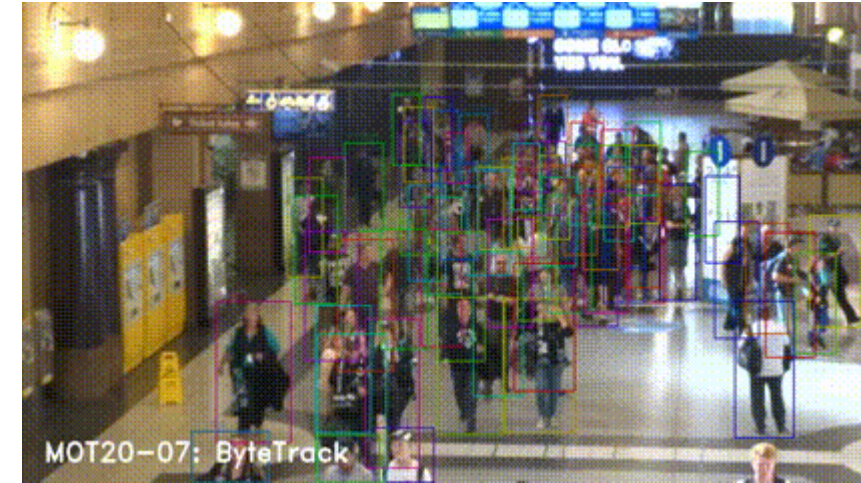
Ziel: Objekt Tracking

Objekt Tracking

- Eindeutige Objekt Detektion von beweglichen Objekten über eine Sequenz von Frames

YOLO hat Objekt Tracking integriert

<https://docs.ultralytics.com/modes/track/>



<https://github.com/ifzhang/ByteTrack>

Sehr gutes Tutorial von Roboflow zu Objekt Tracking mit YOLOv8 (14 Min.):

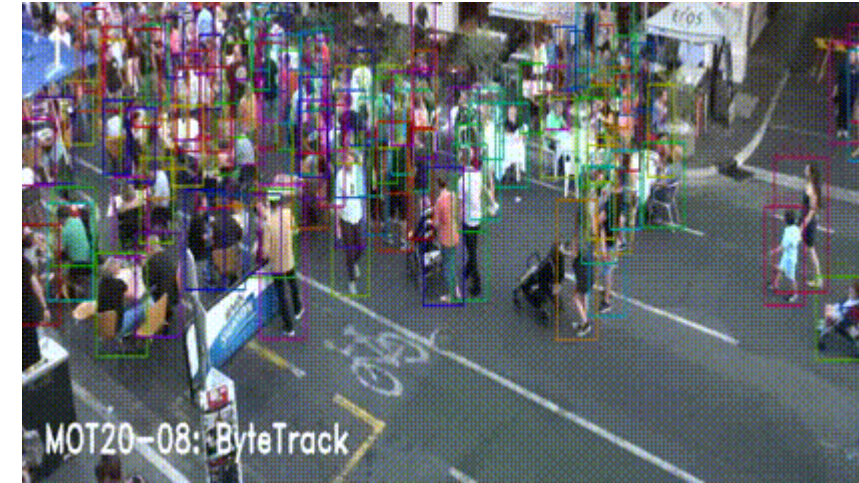
https://www.youtube.com/watch?v=Mt9iHFd0_Bo

Welche Rolle spielt Computer Vision für CPS?

Ziel: Objekt Tracking

Vorgehensweise:

- Tracking-by-Detection
 - Objekt Detektion (bspw. mit YOLO)
 -
 -



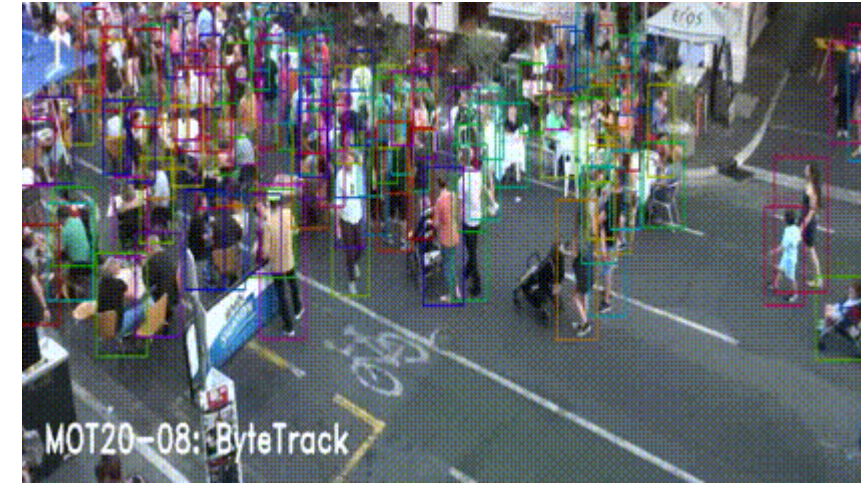
<https://github.com/ifzhang/ByteTrack>

Welche Rolle spielt Computer Vision für CPS?

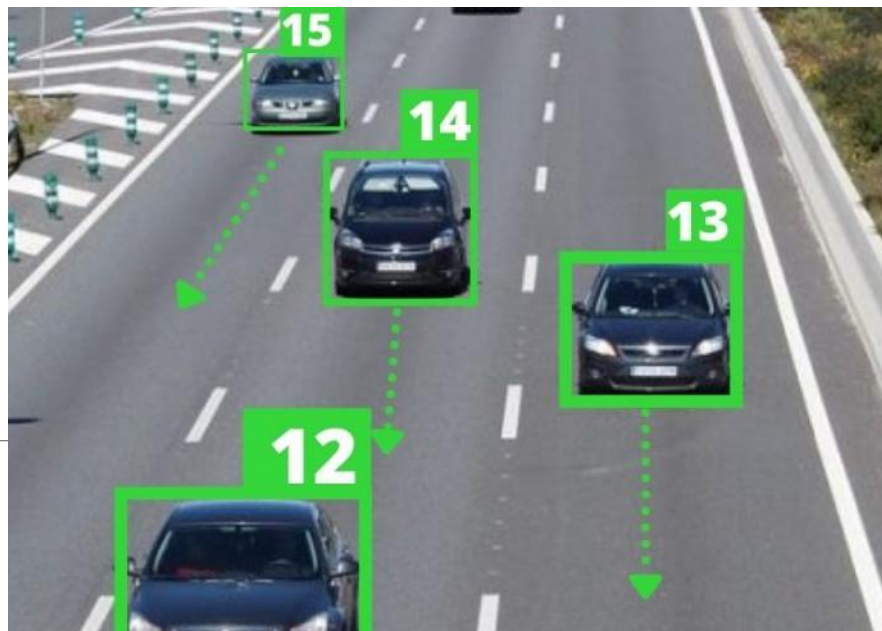
Ziel: Objekt Tracking

Vorgehensweise:

- Tracking-by-Detection
 - Objekt Detektion (bspw. mit YOLO)
 - Tracking der detektierten Objekte
 - Kalman Filter prädiziert wo Objekt in nächstem Frame sein könnte. In dieser Region wird dann für das nächste Frame Objekt Detektion ausgeführt.



<https://github.com/ifzhang/ByteTrack>



13.10.2026

Seite 236

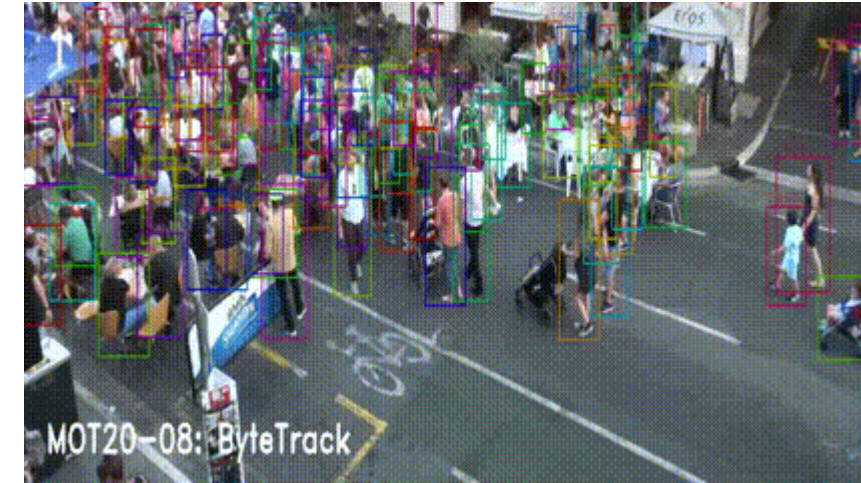
<https://pysource.com/2021/11/02/kalman-filter-predict-the-trajectory-of-an-object/>

Welche Rolle spielt Computer Vision für CPS?

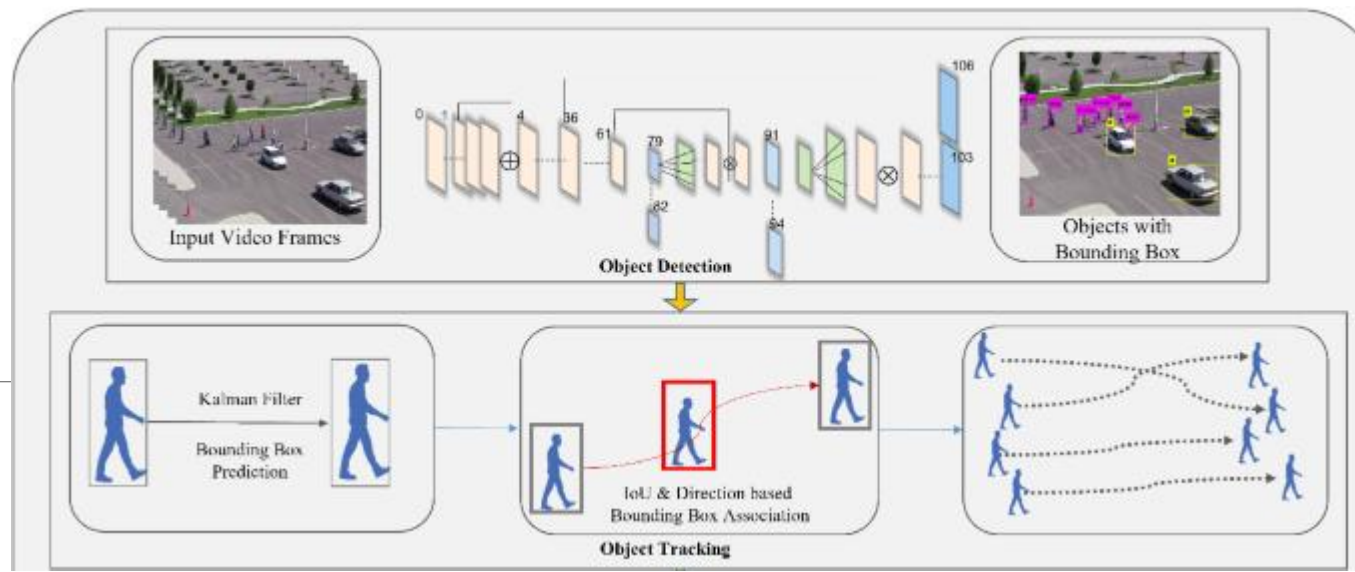
Ziel: Objekt Tracking

Vorgehensweise:

- Tracking-by-Detection
 - Objekt Detektion (bspw. mit YOLO)
 - Tracking der detektierten Objekte
 - Kalman Filter prädiziert wo Objekt in nächstem Frame sein könnte. In dieser Region wird dann für das nächste Frame Objekt Detektion ausgeführt.
 - Re-Identifikation (ist das Objekt noch das gleiche?)



<https://github.com/ifzhang/ByteTrack>



13.10.2026

Seite 237

A. Patel et al., Motion-compensated online object tracking for activity detection and crowd behavior analysis, The visual Computer, 2022

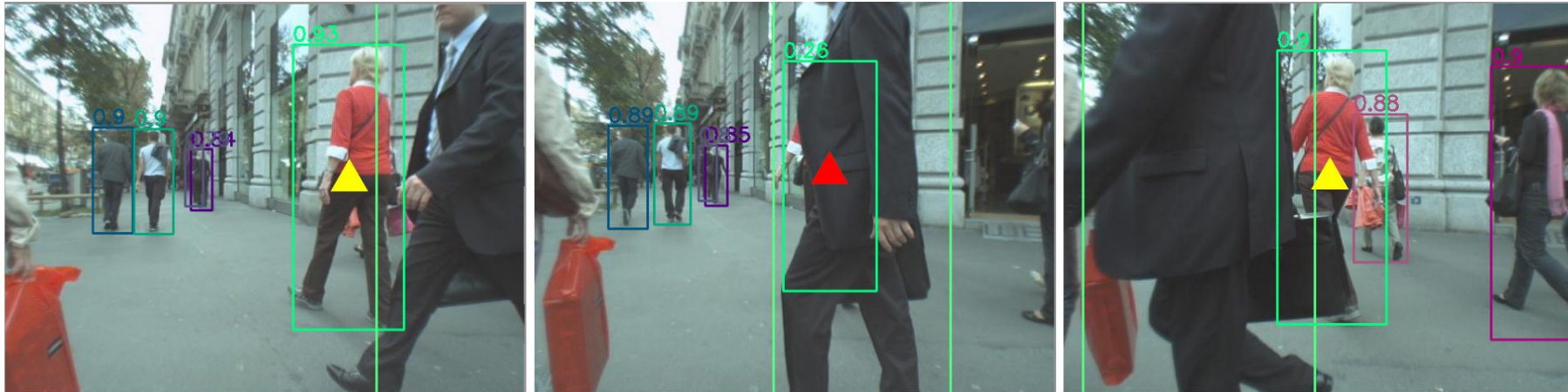
Technology
Arts Sciences
TH Köln

Welche Rolle spielt Computer Vision für CPS?

Ziel: Objekt Tracking

In YOLO integriert („State of the Art“ des Objekt Trackings):

- ByteTrack, <https://github.com/ifzhang/ByteTrack>
- Beibehalten von Bounding Boxen, die temporär nur einen geringen Score erhalten



Y. Zhang et al., ByteTrack: Multi-Object Tracking by Associating Every Detection Box, 2022

▪

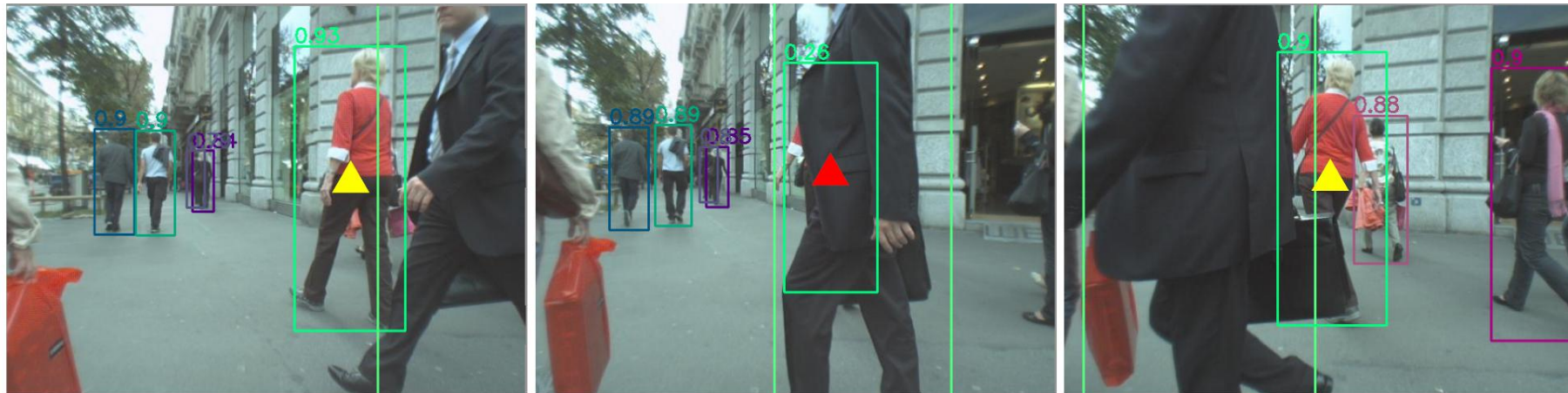
▪

Welche Rolle spielt Computer Vision für CPS?

Ziel: Objekt Tracking

In YOLO integriert („State of the Art“ des Objekt Trackings):

- ByteTrack, <https://github.com/ifzhang/ByteTrack>
- Beibehalten von Bounding Boxen, die temporär nur einen geringen Score erhalten



Y. Zhang et al., ByteTrack: Multi-Object Tracking by Associating Every Detection Box, 2022

- BoT-SORT, <https://github.com/NirAharon/BoT-SORT>
 - Erweiterung von ByteTrack um Kompensation der Bewegung der Kamera, verbesserter Zustandsvektor des Kalman Filters, ...

YOLO26 Objekttracking testen

```
from ultralytics import YOLO
```

```
model = YOLO("yolo26n.pt") # Load an official Detect model
```

```
# Perform tracking with the model with ByteTrack
```

```
results = model.track("https://youtu.be/LNwODJXcvt4", show=True, tracker="bytetrack.yaml")
```

<https://docs.ultralytics.com/modes/track/>

s. Notebook: [02_objekttracking_yolo26.ipynb](#)

Welche Rolle spielt Computer Vision für CPS?

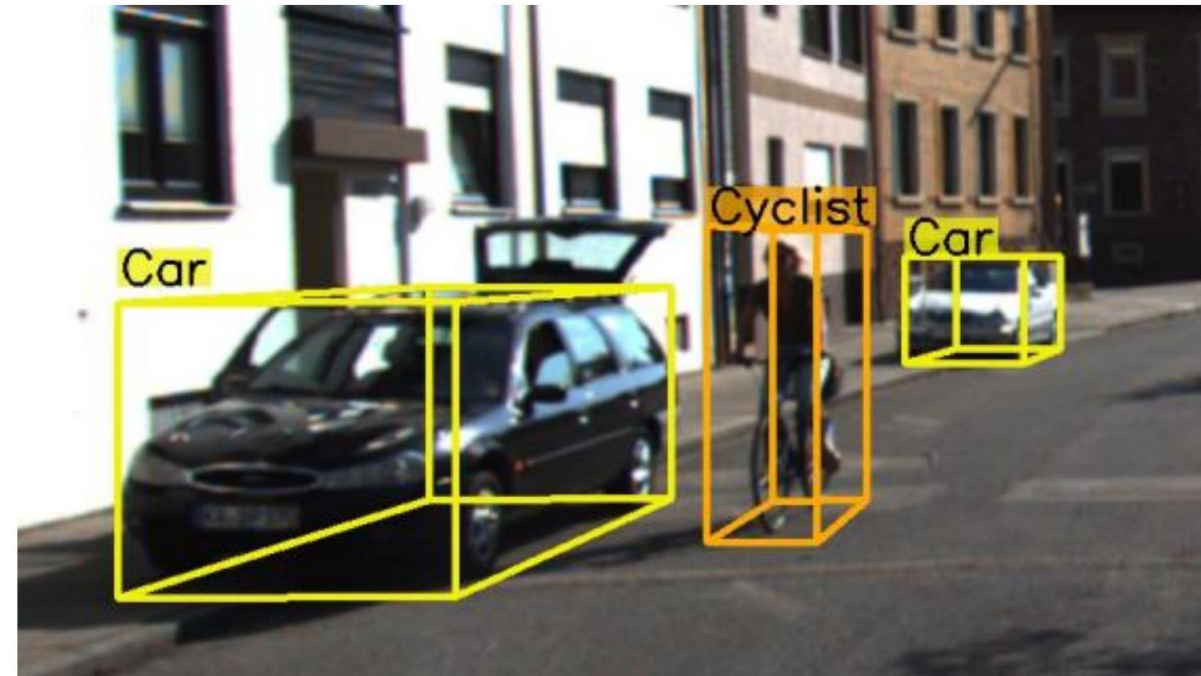
Ziel: 3D Objekt Tracking (Detektion)

Mit Lidar (oder Radar) Daten können Sie Objekte als 3D Objekte tracken/detektieren

- K. Chen et al., MVLidarNet: Real-Time Multi-Class Scene Understanding for Autonomous Driving using multiple views, 2020
- Popov et al., NV RadarNet: Real-Time Radar Obstacle and Free Space Detection for Autonomous Driving, 2023

Bibliotheken:

- github.com/open-mmlab/mmdetection3d
- github.com/isl-org/Open3D-ML



Welche Rolle spielt Computer Vision für CPS?

Ziel: Objekt Detektion von beliebigen Objektkategorien

- Algorithmen der Objekt Detektion funktionieren nur für die Objektkategorien für die sie trainiert wurden
- YOLO wurde trainiert mit COCO Datensatz
 - 80 verschiedene Objektkategorien (Autos, Fahrräder, Tiere, Regenschirme, ...)

▪

▪

▪

▪

▪

▪

Welche Rolle spielt Computer Vision für CPS?

Ziel: Objekt Detektion von beliebigen Objektkategorien

- Algorithmen der Objekt Detektion funktionieren nur für die Objektkategorien für die sie trainiert wurden
- YOLO wurde trainiert mit COCO Datensatz
 - 80 verschiedene Objektkategorien (Autos, Fahrräder, Tiere, Regenschirme, ...)
- Mit Trainingsdaten von weiteren Objektkategorien können Sie auch diese detektieren
- -
 -
 -
 -
 -

Welche Rolle spielt Computer Vision für CPS?

Ziel: Objekt Detektion von beliebigen Objektkategorien

- Algorithmen der Objekt Detektion funktionieren nur für die Objektkategorien für die sie trainiert wurden
- YOLO wurde trainiert mit COCO Datensatz
 - 80 verschiedene Objektkategorien (Autos, Fahrräder, Tiere, Regenschirme, ...)
- Mit Trainingsdaten von weiteren Objektkategorien können Sie auch diese detektieren
- Was ist, wenn Sie „beliebige“ Objektkategorien detektieren möchten?
 - Open-Vocabulary Objekt Detektion
 - Trainiert mit Millionen von Bildern aus dem Internet und deren Bildbeschriftungen
- YOLOE - Real-Time Seeing Anything
 - <https://docs.ultralytics.com/models/yoloe>

Welche Rolle spielt Computer Vision für CPS?

Ziel: Objekt Detektion von beliebigen Objektkategorien

- Open-Vocabulary
Objekt Detektion
- <https://github.com/ailab-cvc/yolo-world>



Welche Rolle spielt Computer Vision für CPS?

Ziel: Objekt Detektion von beliebigen Objektkategorien

- Open-Vocabulary Objekt Detektion
- <https://github.com/ailab-cvc/yolo-world>



Welche Rolle spielt Computer Vision für CPS?

Ziel: Objekt Detektion von beliebigen Objektkategorien

```
from ultralytics import YOLO
```

```
model = YOLO("yoloe-11s-seg.pt")           # Initialize a model
```

```
model.set_classes(["person", "bus"])        # Define custom classes
```

```
results = model.predict("path/to/image.jpg") # Run inference to detect your custom classes
```

```
results[0].show()                          # Show results
```

Welche Rolle spielt Computer Vision für CPS?

Ziel: Objekt Detektion von beliebigen Objektkategorien

```
from ultralytics import YOLO
```

```
model = YOLO("yoloe-11s-seg.pt")           # Initialize a model
```

```
model.set_classes(["person", "bus"])        # Define custom classes
```

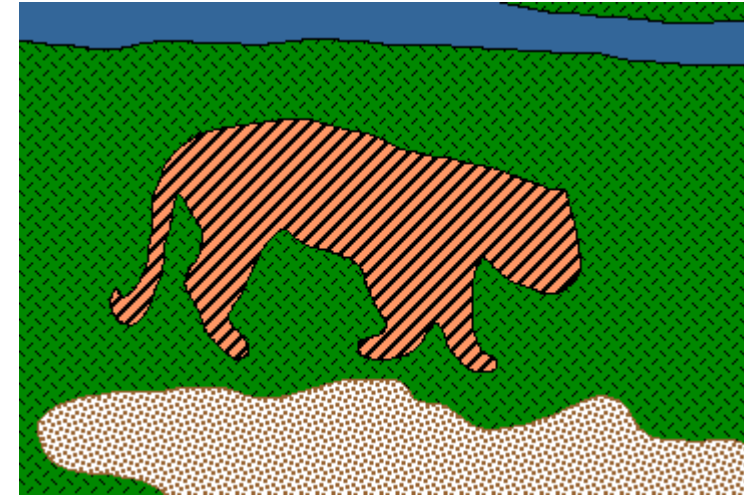
```
results = model.predict("path/to/image.jpg") # Run inference to detect your custom classes
```

```
results[0].show()                           # Show results
```

s. Notebook [03_AppObjectDetection.ipynb](#)

Welche Rolle spielt Computer Vision für CPS?

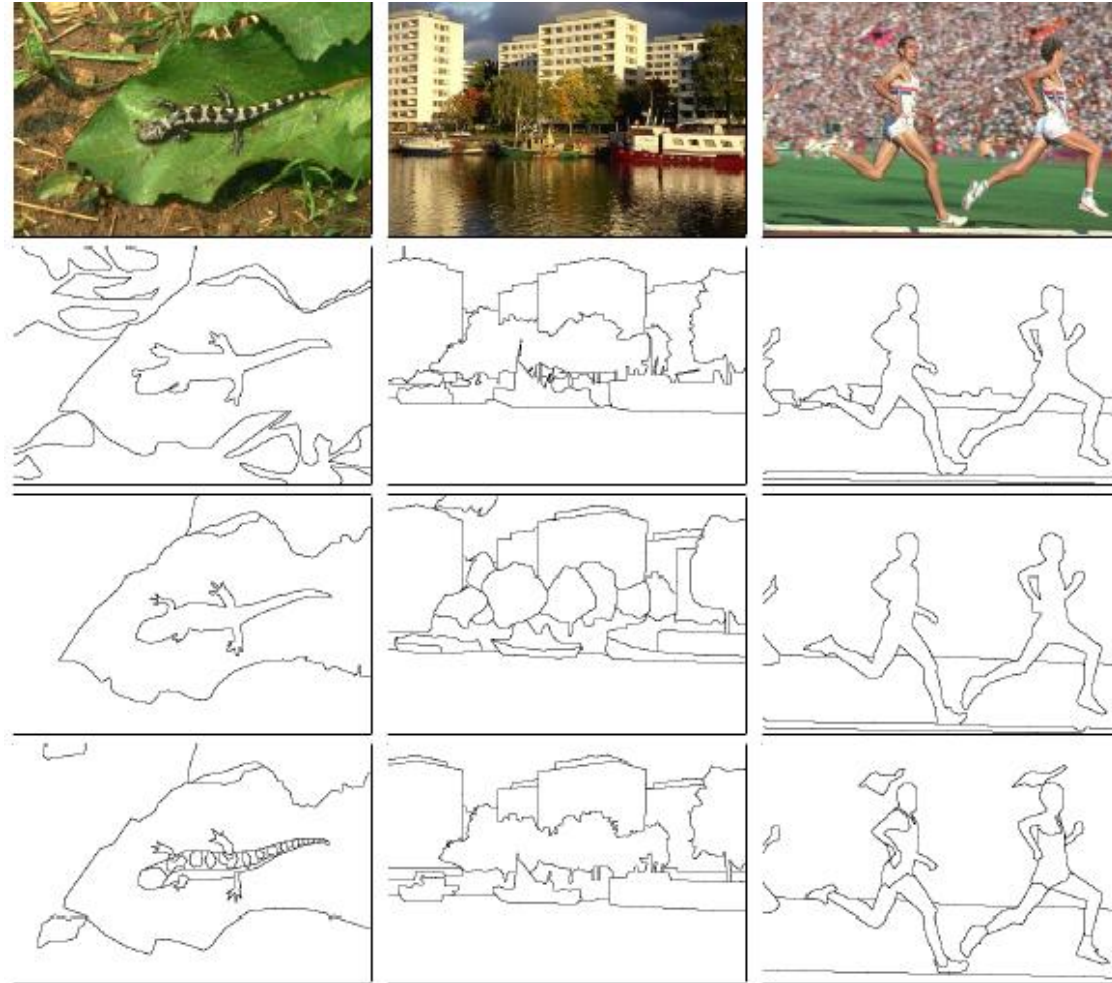
Ziel: Bildsegmentierung



Welche Rolle spielt Computer Vision für CPS?

Ziel: Bildsegmentierung

- Es gibt keine objektive Definition einer „richtigen“ Segmentierung!
- Segmentierung wird durch Menschen vorgegeben



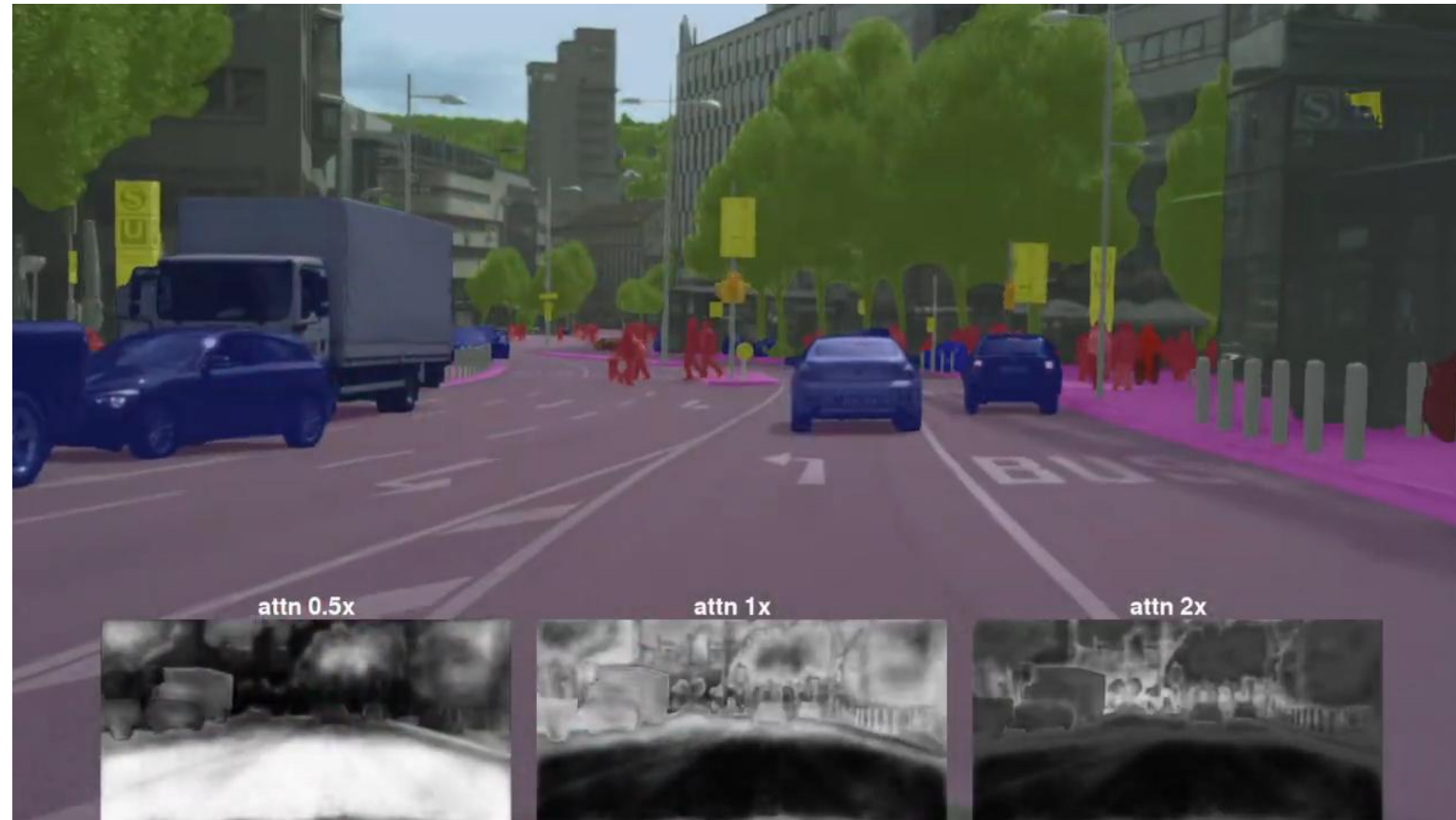
Subject 1

Subject 2

Subject 3

Welche Rolle spielt Computer Vision für CPS?

Ziel: Bildsegmentierung



Bildsegmentierung mit Deep Learning

Trainingsdaten

Eingangsdaten



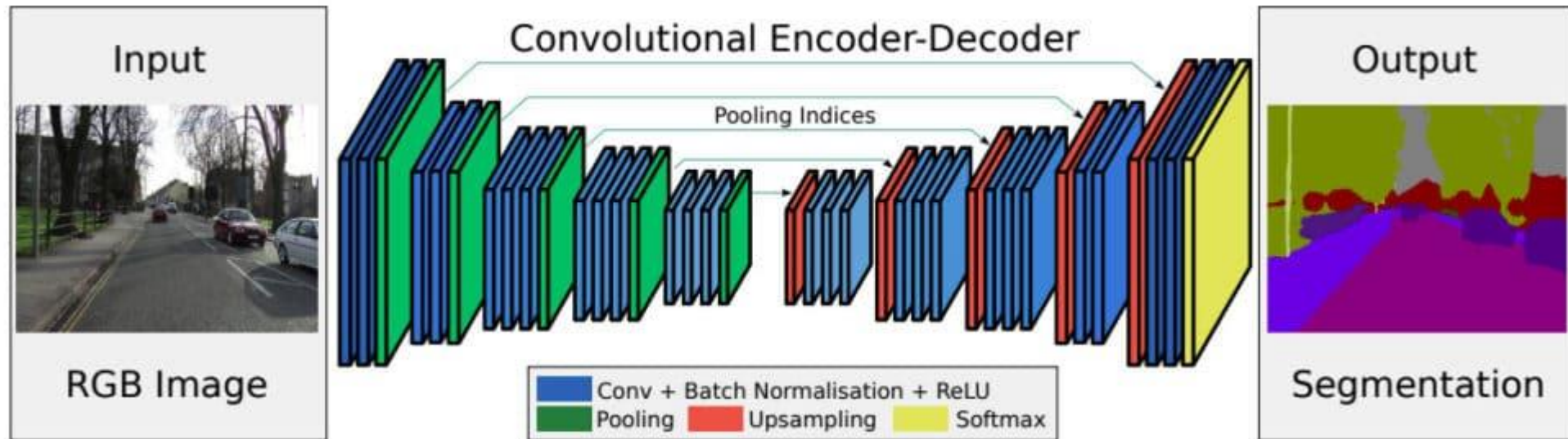
Ausgangsdaten



Quelle für Trainingsdaten: <https://paperswithcode.com/datasets?task=semantic-segmentation> (261 Datensätze)

Bildsegmentierung mit Deep Learning

Training des Deep Learning Modells (bspw. Autoencoder)



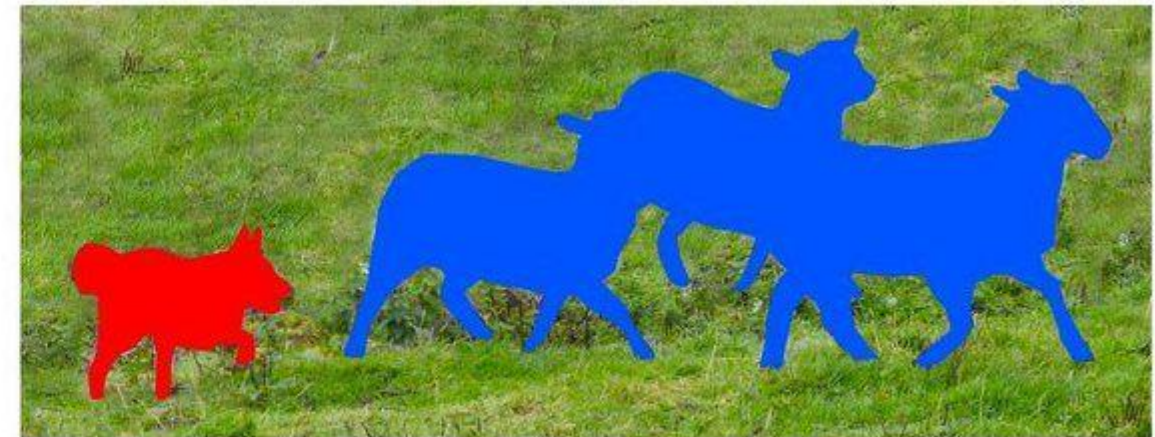
Bildsegmentierung mit Deep Learning

Instance Segmentation vs. Semantic Segmentation

<https://blog.roboflow.com/difference-semantic-segmentation-instance-segmentation/>



Image Recognition



Semantic Segmentation

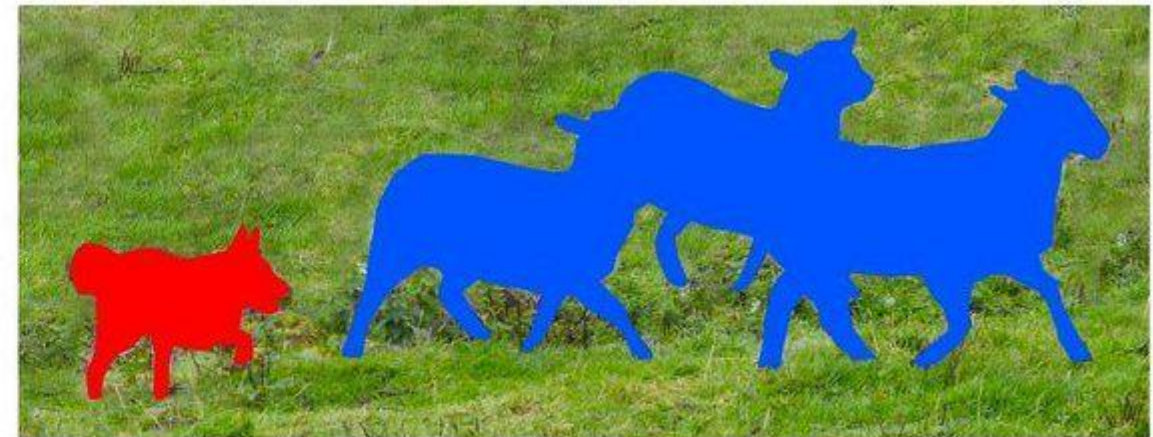
Bildsegmentierung mit Deep Learning

Instance Segmentation vs. Semantic Segmentation

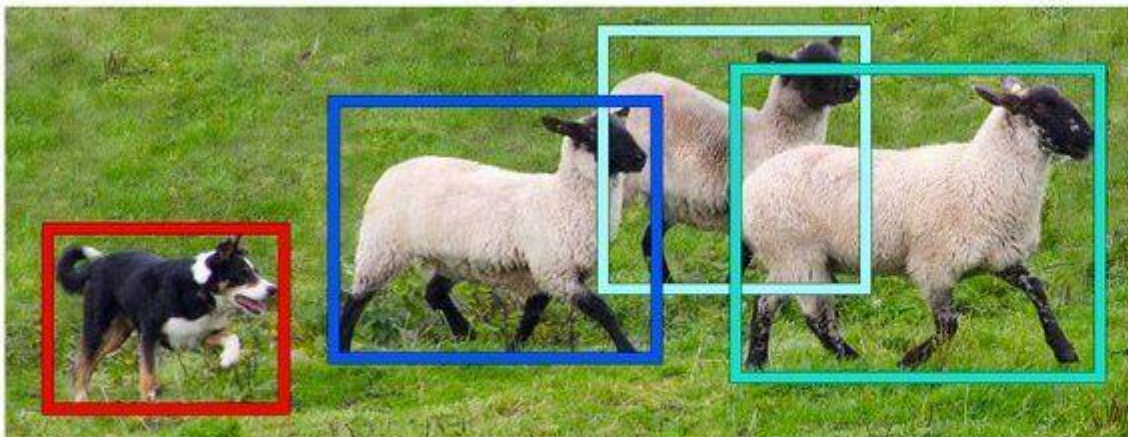
<https://blog.roboflow.com/difference-semantic-segmentation-instance-segmentation/>



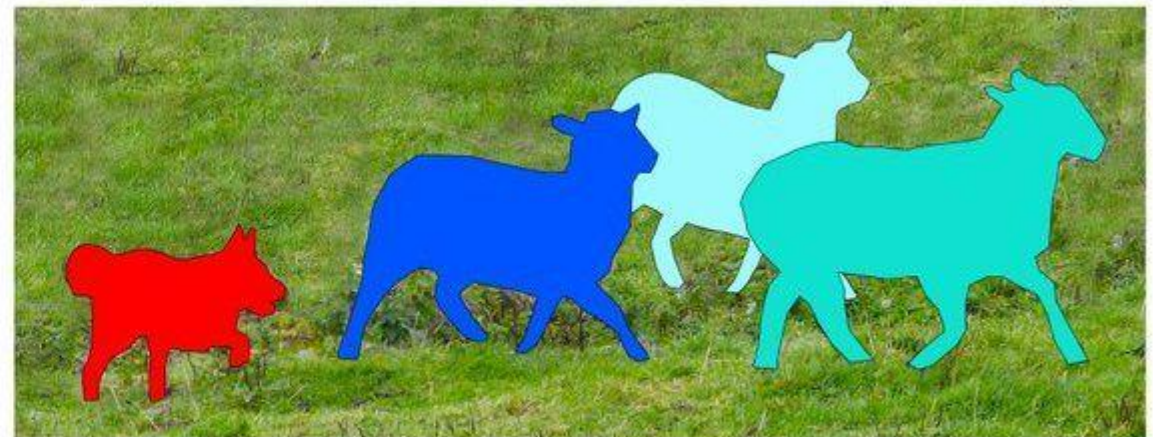
Image Recognition



Semantic Segmentation



Object Detection



Instance Segmentation

Bildsegmentierung mit Deep Learning

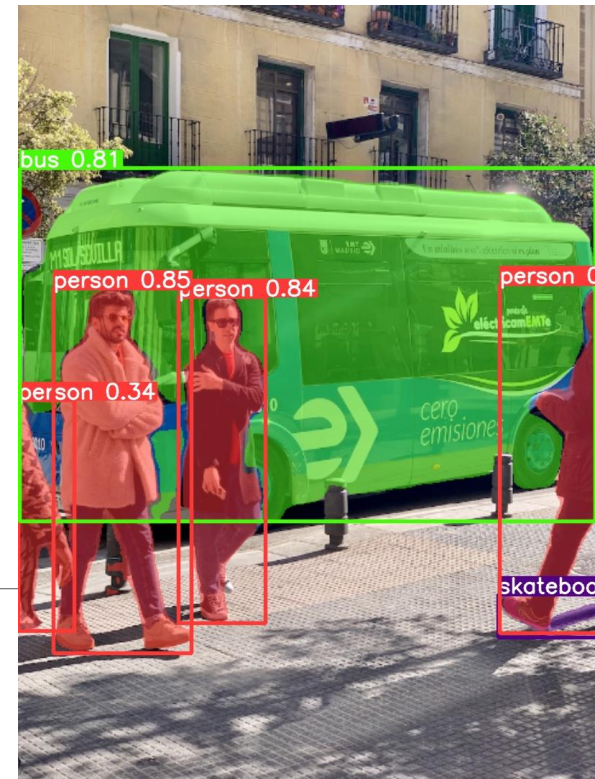
YOLO26 kann auch Instance Segmentation (<https://docs.ultralytics.com/tasks/segment/>)

Gutes Tutorial zu YOLOv8 zu Image Segmentation und Objekt Detektion, nur Inferenz (13 Min.)

<https://www.youtube.com/watch?v=75LI9MI9eEo>

```
from ultralytics import YOLO  
model = YOLO("yolo26n-seg.pt") # load a model  
# Predict with the model  
results = model("https://ultralytics.com/images/bus.jpg")
```

s. Notebook: [06_objektdetektion_yolo26.ipynb](#)



Welche Rolle spielt Computer Vision für CPS?

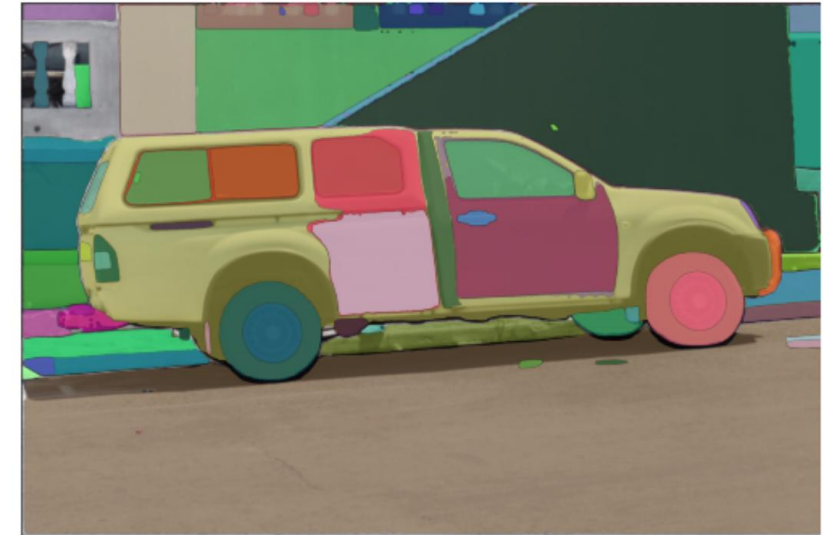
Bildsegmentierung: Instance Segmentation mit YOLOv8



Open-Vocabulary Image Segmentation

Segment Anything Modell (SAM 3)

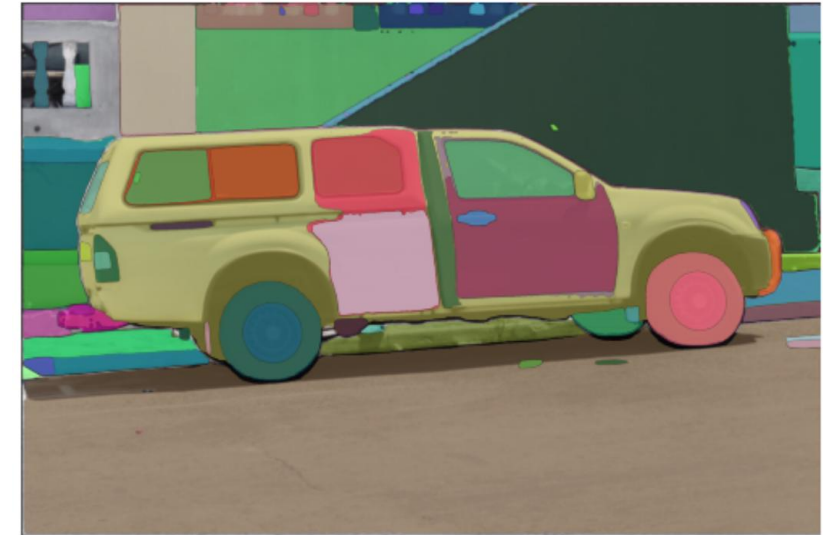
- github.com/facebookresearch/segment-anything
- <https://docs.ultralytics.com/models/sam-3>
- Foundation Modell für Bildsegmentierung; Promptable
- FastSAM (<https://docs.ultralytics.com/models/fast-sam/>)
 - YOLOv8 Segmentierungsmodell und anschließend Fokus auf die gewünschten Begriffe



Open-Vocabulary Image Segmentation

Segment Anything Modell (SAM 3)

- github.com/facebookresearch/segment-anything
- <https://docs.ultralytics.com/models/sam-3>
- Foundation Modell für Bildsegmentierung; Promptable
- FastSAM (<https://docs.ultralytics.com/models/fast-sam/>)
 - YOLOv8 Segmentierungsmodell und anschließend Fokus auf die gewünschten Begriffe
- Grounded Segment Anything
 - Grounding DINO (Open-Vocabulary Objekt Detektion) – lokalisiert die Objekte; geg. Prompt
 - <https://github.com/IDEA-Research/Grounding-DINO-1.5-API>
 - Segment Anything Modell – segmentiert die lokalisierten Objekte
 - <https://github.com/IDEA-Research/Grounded-SAM-2>



Open-Vocabulary Image Segmentation

```
from ultralytics import FastSAM
```

```
# Define an inference source
```

```
source = "path/to/bus.jpg"
```

```
# Create a FastSAM model
```

```
model = FastSAM("FastSAM-s.pt") # or FastSAM-x.pt
```

```
# Run inference on an image
```

```
everything_results = model(source, device="cpu", retina_masks=True, imgsz=1024,  
conf=0.4, iou=0.9)
```

```
# Run inference with texts prompt
```

```
results = model(source, texts="a photo of a dog")
```

[Jupyter Notebook
zu FastSAM](#)

Multimodale Large Language Modelle: Meta Llama4

Which tools in the image can be used for measuring length?



Provide bounding boxes for every recognized item.



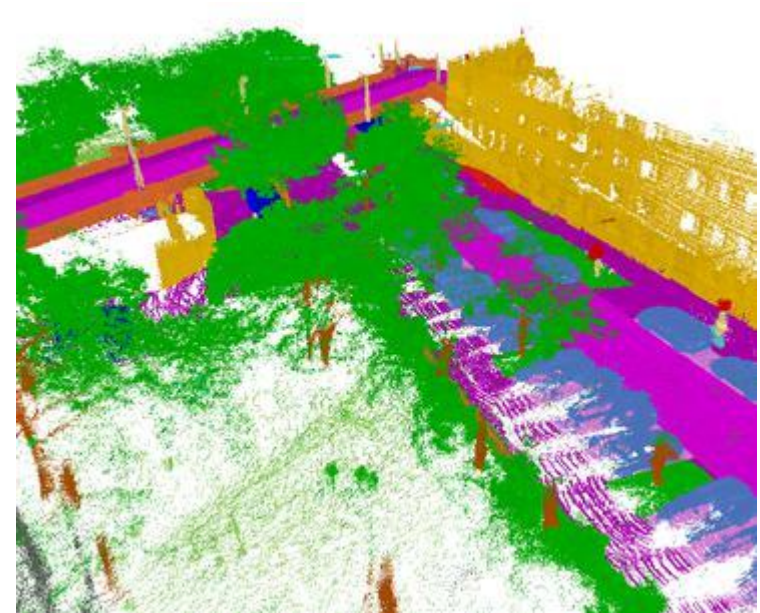
3D Bildsegmentierung

Segmentierung der Punktwolken von Lidar (oder Radar) Sensordaten

<https://github.com/isl-org/Open3D-ML>

Bildquelle:

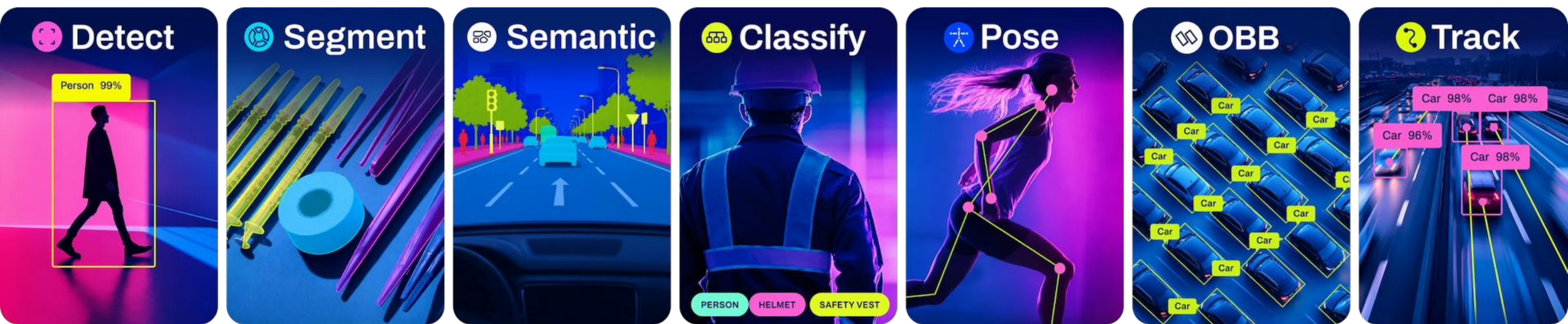
<https://paperswithcode.com/task/3d-semantic-segmentation>



Objektdetektion, -tracking und -segmentierung

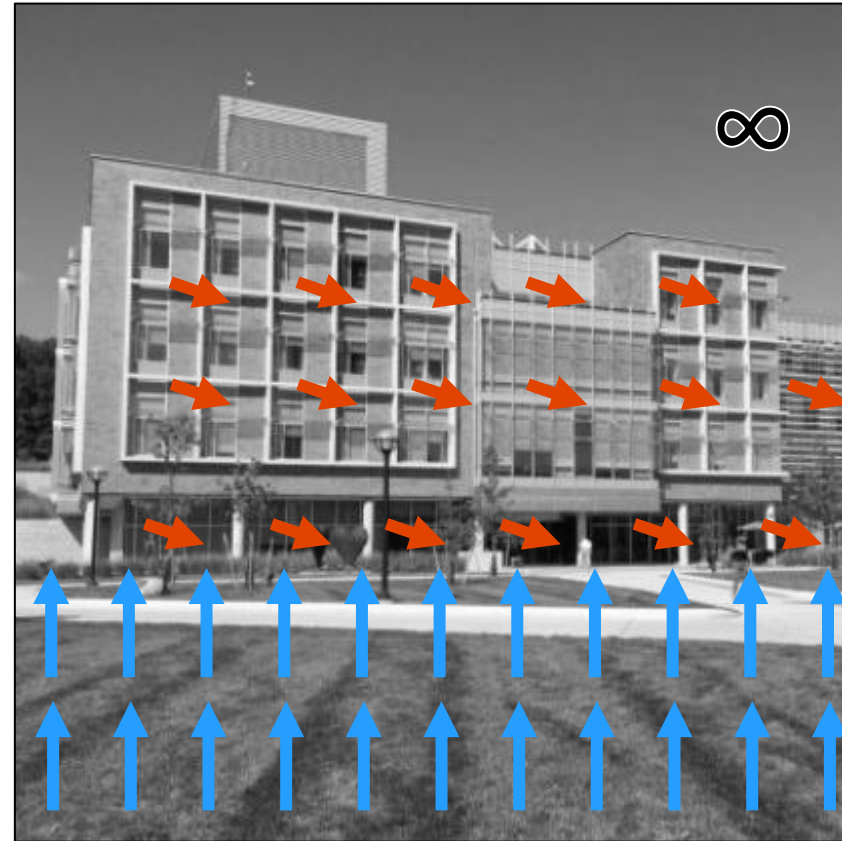
Zusammenfassung

- Mit YOLO-Modellen von Ultralytics kann man viele Bilderkennungs-Anwendungen recht einfach starten



Welche Rolle spielt Computer Vision für CPS?

Ziel: 3D Struktur (Oberflächennormale)



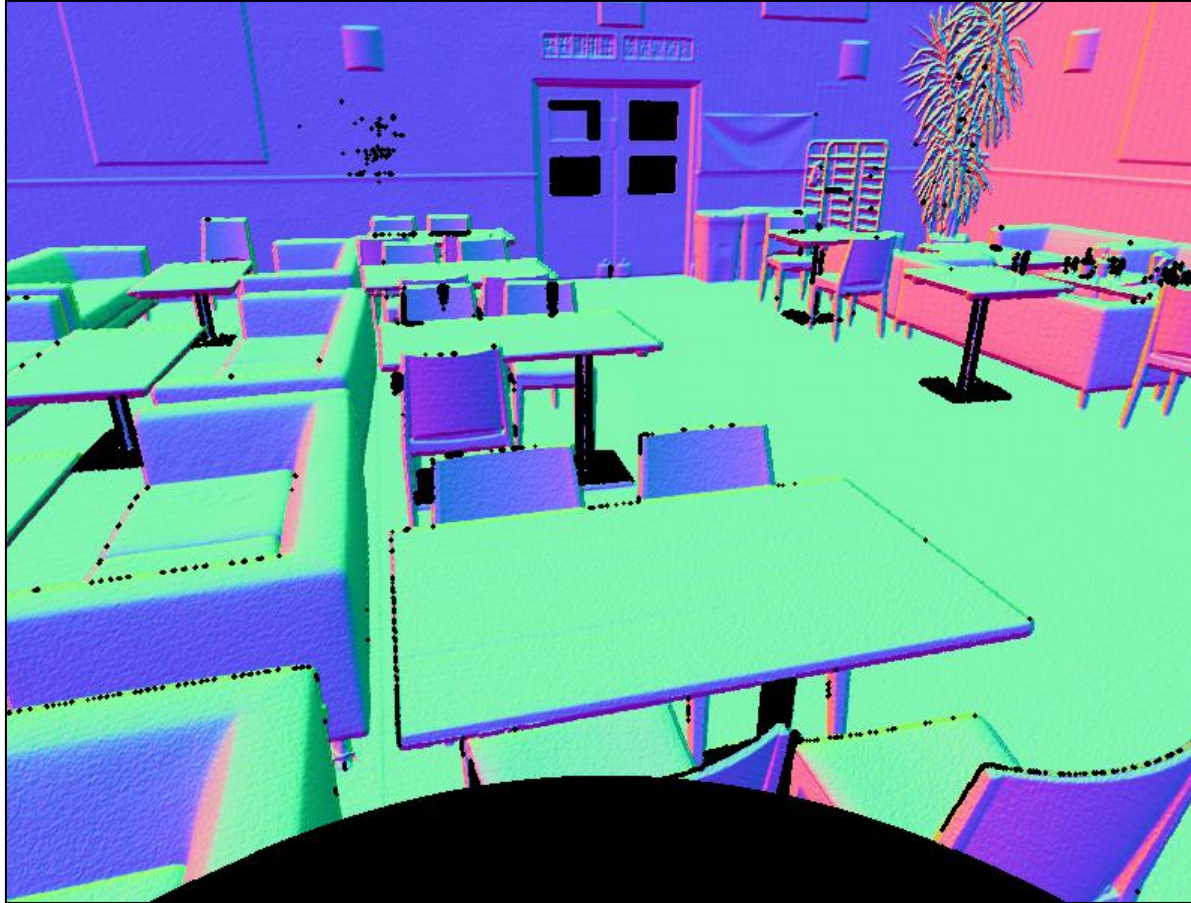
Welche Rolle spielt Computer Vision für CPS?

Ziel: 3D Struktur (Oberflächennormale)

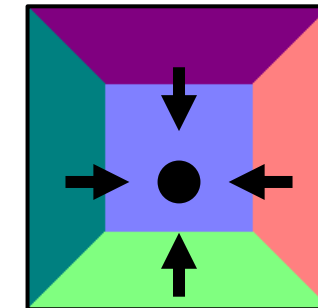


Welche Rolle spielt Computer Vision für CPS?

Ziel: 3D Struktur (Oberflächennormale)



Raum



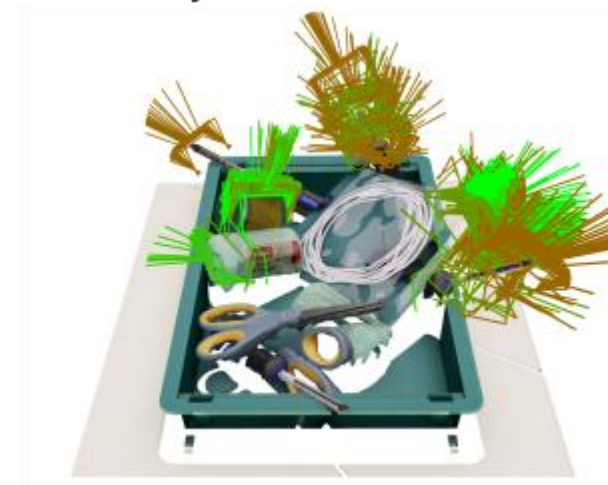
Legende

Welche Rolle spielt Computer Vision für CPS?

Ziel: Finden von Greifpunkten

Bin Picking

- Finden von Greifpunkten an Objekten



Inhalt

Modellierung der Umgebung von CPS:

- Modellierung mit KI/Deep Learning
 - Objektdetektion, -tracking, Bildsegmentierung, Greifpunkte
 - **Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?**
 - Spracherkennung
 - Verhaltensmodellierung von Fußgängern, Fahrzeugen
 - World Models
- Lernen in virtuellen Umgebungen (in einer Simulation)
 - 3D Simulation
 - Von der Simulation in die Realität (Sim2real)
- Statische Modelle der Umgebung (Karte)
 - SLAM – Simultaneous Localization and Mapping
 - Semantische Karte (Karte mit Objekten/Orten: Couch, TV, Küche, ...)

Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?



Blickwinkel Variation

Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?



Blickwinkel Variation

Beleuchtung



Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?



Blickwinkel Variation

Beleuchtung



Credit: Flickr user michaelpaul

Skalierung

Quelle: Intro to Computer Vision (CS5670),
Cornell Tech, Spring 2021

Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?



Variation innerhalb einer Klasse

Quelle: Intro to Computer Vision (CS5670),
Cornell Tech, Spring 2021

Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?



Variation innerhalb einer Klasse



Bewegung (Source: S. Lazebnik)

Quelle: Intro to Computer Vision (CS5670),
Cornell Tech, Spring 2021

Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?



Variation innerhalb einer Klasse



Bewegung (Source: S. Lazebnik)

Background clutter



Quelle: Intro to Computer Vision (CS5670),
Cornell Tech, Spring 2021

13.10.2026

Seite 274

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?



Variation innerhalb einer Klasse



Bewegung (Source: S. Lazebnik)

Background clutter



Quelle: Intro to Computer Vision (CS5670),
Cornell Tech, Spring 2021

Occlusion

13.10.2026

Seite 275

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

Wettbewerb: “Überlisten Sie YOLO”

Teilen Sie sich in Teams auf und testen Sie YOLO auf dem Smartphone.

Das Team, das die meisten Fehler provoziert, gewinnt. Es zählen:

- ein Objekt, das fälschlich als “Person” erkannt wird,
- eine Person, die nicht erkannt wird (Verdeckung, Pose, Verkleidung),
- ein Phantomobjekt (Spiegelung, Plakat, Foto eines Objekts).





13.10.2026

Seite 277

Quelle: Computer Vision (CS 4476),
Georgia Tech, Fall 2019

ology
sciences
In



13.10.2026

Seite 278

Quelle: Computer Vision (CS 4476),
Georgia Tech, Fall 2019

Technology
Sciences
öln



Quelle: Computer Vision (CS 4476),
Georgia Tech, Fall 2019

13.10.2026

Seite 279



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

Quelle: Computer Vision (CS 4476), Georgia Tech, Fall 2019

Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig? - Unterm Strich

Wahrnehmung ist von Natur aus ein mehrdeutiges Problem

- Viele verschiedene 3D-Szenen können zu einem bestimmten 2D-Bild führen.

■

■



Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig? - Unterm Strich

Wahrnehmung ist von Natur aus ein mehrdeutiges Problem

- Viele verschiedene 3D-Szenen können zu einem bestimmten 2D-Bild führen.
- Wir müssen oft auf Vorwissen über die Struktur der Welt zurückgreifen

■



Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig? - Unterm Strich

Wahrnehmung ist von Natur aus ein mehrdeutiges Problem

- Viele verschiedene 3D-Szenen können zu einem bestimmten 2D-Bild führen.
- Wir müssen oft auf Vorwissen über die Struktur der Welt zurückgreifen
- Deep Learning Modelle haben das „Vorwissen über die Struktur der Welt“ aus Millionen von Trainingsbildern erlernt



Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig? - Unterm Strich

Was denken Sie antwortet GPT-4vision?



Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig? - Unterm Strich

Was denken Sie antwortet GPT-4vision?

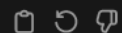


ChatGPT

Das Bild zeigt eine Straßenmalerei eines Coca-Cola-Flaschenbildes, das so gestaltet ist, dass es, aus einer bestimmten Perspektive betrachtet, dreidimensional erscheint. Dies ist ein Beispiel für eine anamorphe Illusion, bei der durch die Verzerrung der Zeichnung und die Wahl des Standpunktes der Kamera eine optische Täuschung erzeugt wird, die Tiefe und Volumen suggeriert, wo tatsächlich nur eine flache Oberfläche vorhanden ist.

Die Person im Bild kniet neben der Straßenmalerei und lächelt in die Kamera, was den Eindruck erweckt, als würde sie neben einer echten, großen Flasche sitzen. Sie scheint die Illusion zu genießen und ist Teil der Inszenierung, die dazu dient, die Täuschung für den Betrachter zu verstärken.

Die Kreiden, die neben dem Bild auf dem Boden liegen, deuten darauf hin, dass die Malerei kürzlich fertiggestellt wurde oder dass der Künstler seine Arbeit noch präsentiert. Solche Straßenkunstwerke sind oft temporär und können als eine Form von öffentlicher Kunst betrachtet werden, die Passanten überrascht und erfreut.



Inhalt

Modellierung der Umgebung von CPS:

- Modellierung mit KI/Deep Learning
 - Objektdetektion, -tracking, Bildsegmentierung, Greifpunkte
 - Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?
 - **Spracherkennung**
 - Verhaltensmodellierung von Fußgängern, Fahrzeugen
 - World Models
- Lernen in virtuellen Umgebungen (in einer Simulation)
 - 3D Simulation
 - Von der Simulation in die Realität (Sim2real)
- Statische Modelle der Umgebung (Karte)
 - SLAM – Simultaneous Localization and Mapping
 - Semantische Karte (Karte mit Objekten/Orten: Couch, TV, Küche, ...)

Einsatz von Spracherkennung und Sound Classification in CPS

Spracherkennung in CPS:

- Steuerung von Robotern und Maschinen durch Sprachbefehle.
- Interaktion mit autonomen Systemen (z.B. Sprachsteuerung in Smart Homes oder Fabriken).
- Verbesserung der Mensch-Maschine-Interaktion durch natürliche Sprache.

▪

▪

▪

Einsatz von Spracherkennung und Sound Classification in CPS

Spracherkennung in CPS:

- Steuerung von Robotern und Maschinen durch Sprachbefehle.
- Interaktion mit autonomen Systemen (z.B. Sprachsteuerung in Smart Homes oder Fabriken).
- Verbesserung der Mensch-Maschine-Interaktion durch natürliche Sprache.

Sound Classification:

- Erkennung von Umgebungsgeräuschen zur Reaktion auf sicherheitskritische Ereignisse (z.B. Maschinenlärm, Alarmer, Warnsignale).
- Ziel: Unfallprävention (z.B. Maschinenstopps bei anomalen Geräuschen).

▪

Einsatz von Spracherkennung und Sound Classification in CPS

Spracherkennung in CPS:

- Steuerung von Robotern und Maschinen durch Sprachbefehle.
- Interaktion mit autonomen Systemen (z.B. Sprachsteuerung in Smart Homes oder Fabriken).
- Verbesserung der Mensch-Maschine-Interaktion durch natürliche Sprache.

Sound Classification:

- Erkennung von Umgebungsgeräuschen zur Reaktion auf sicherheitskritische Ereignisse (z.B. Maschinenlärm, Alarmer, Warnsignale).
- Ziel: Unfallprävention (z.B. Maschinenstopps bei anomalen Geräuschen).

Emotionserkennung:

- Mensch-Roboter-Interaktion durch Erkennung emotionaler Zustände verbessern (z.B. Roboter im Gesundheitswesen/Therapie, die auf emotionale Signale reagieren).

Aktueller Stand der Forschung – Spracherkennung

Deep Learning und neuronale Netze:

- Moderne Spracherkennungssysteme basieren auf tiefen neuronalen Netzen wie Transformer-Modellen (z.B. OpenAI's Whisper v3, huggingface.co/openai/whisper-large-v3).
- Große Fortschritte durch vortrainierte Modelle, die natürliche Sprache verstehen und interpretieren.
-
-

Aktueller Stand der Forschung – Spracherkennung

Deep Learning und neuronale Netze:

- Moderne Spracherkennungssysteme basieren auf tiefen neuronalen Netzen wie Transformer-Modellen (z.B. OpenAI's Whisper v3, huggingface.co/openai/whisper-large-v3).
- Große Fortschritte durch vortrainierte Modelle, die natürliche Sprache verstehen und interpretieren.

Verbesserungen bei der Genauigkeit:

- Forschung fokussiert sich auf die Reduzierung von Fehlern in lauten Umgebungen und Multilingualität.
- Verwendung von Self-Supervised Learning zur Verarbeitung großer Mengen unbeschrifteter Daten.

Aktueller Stand der Forschung – Spracherkennung

Deep Learning und neuronale Netze:

- Moderne Spracherkennungssysteme basieren auf tiefen neuronalen Netzen wie Transformer-Modellen (z.B. OpenAI's Whisper v3, huggingface.co/openai/whisper-large-v3).
- Große Fortschritte durch vortrainierte Modelle, die natürliche Sprache verstehen und interpretieren.

Verbesserungen bei der Genauigkeit:

- Forschung fokussiert sich auf die Reduzierung von Fehlern in lauten Umgebungen und Multilingualität.
- Verwendung von Self-Supervised Learning zur Verarbeitung großer Mengen unbeschrifteter Daten.

<https://dgaida.github.io/speech2text/dev/>

Demo auf GPU-Laptop

Aktueller Stand der Forschung – Keyword Detection

Keyword Spotting (KWS):

- Erkennung von Schlüsselwörtern („Wake Words“) wie „Hey Siri“ oder „Alexa“ durch sehr kleine CNNs zur Ausführung auf eingebetteter Hardware.

-

-

-

Aktueller Stand der Forschung – Keyword Detection

Keyword Spotting (KWS):

- Erkennung von Schlüsselwörtern („Wake Words“) wie „Hey Siri“ oder „Alexa“ durch sehr kleine CNNs zur Ausführung auf eingebetteter Hardware.

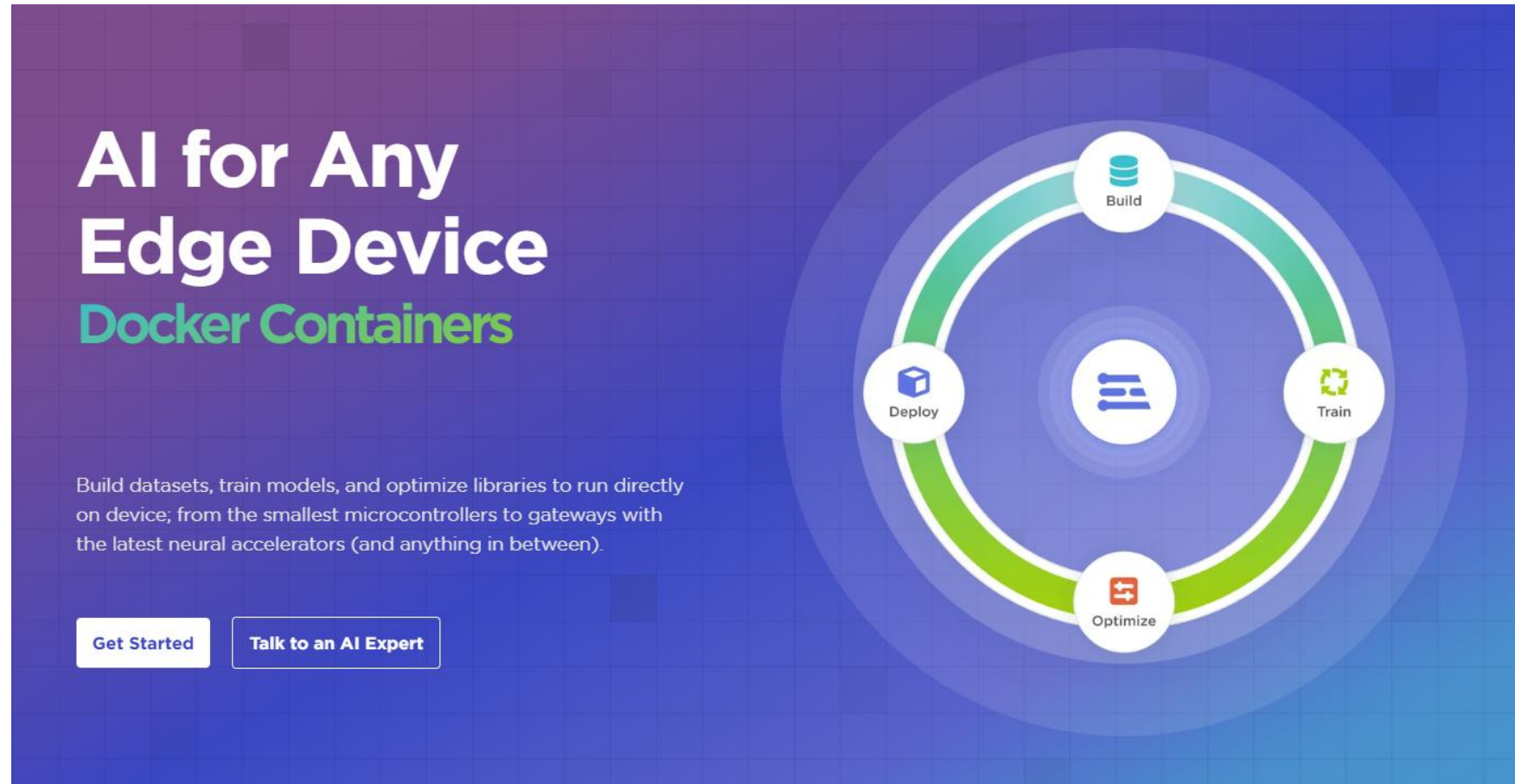
Forschungsansätze:

- Entwicklung von kleinen Modellen mit geringem Speicher- und Energieverbrauch (TinyML).
- Steigerung der Genauigkeit durch Federated Learning, das verteilte Datenverarbeitung erlaubt, ohne zentrale Datenpools zu benötigen.
- [Keyword Spotting with the NDP120-Powered Arduino Nicla Voice](#) (Edge Impulse, Youtube)

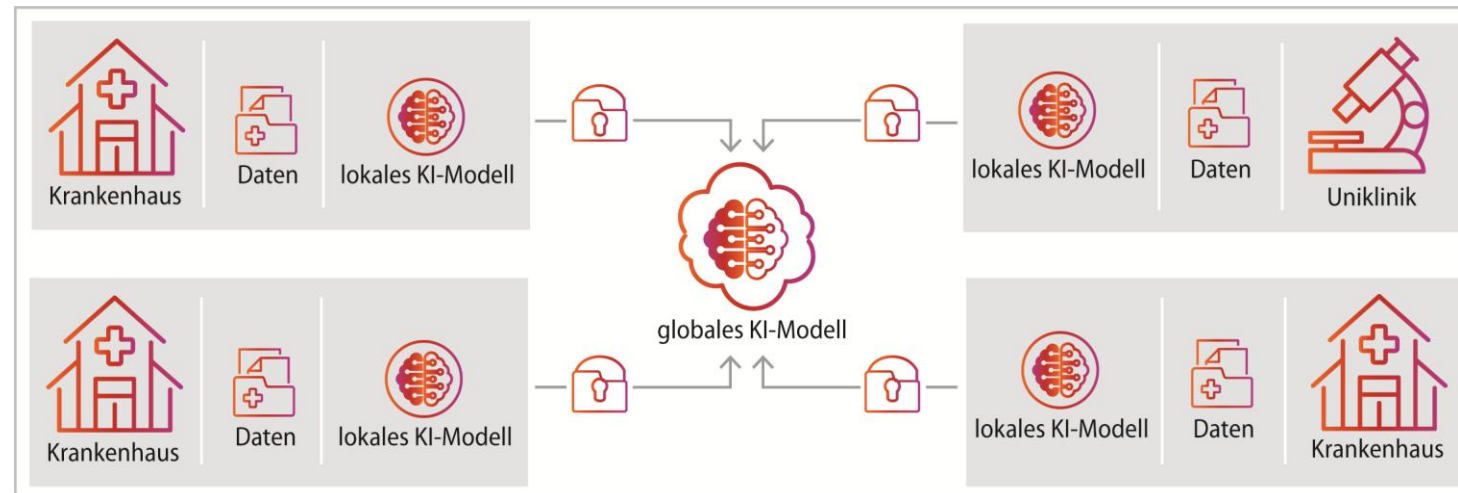


Arduino Pro - Nicla Voice

Edge Impulse



Federated Learning



Inhalt

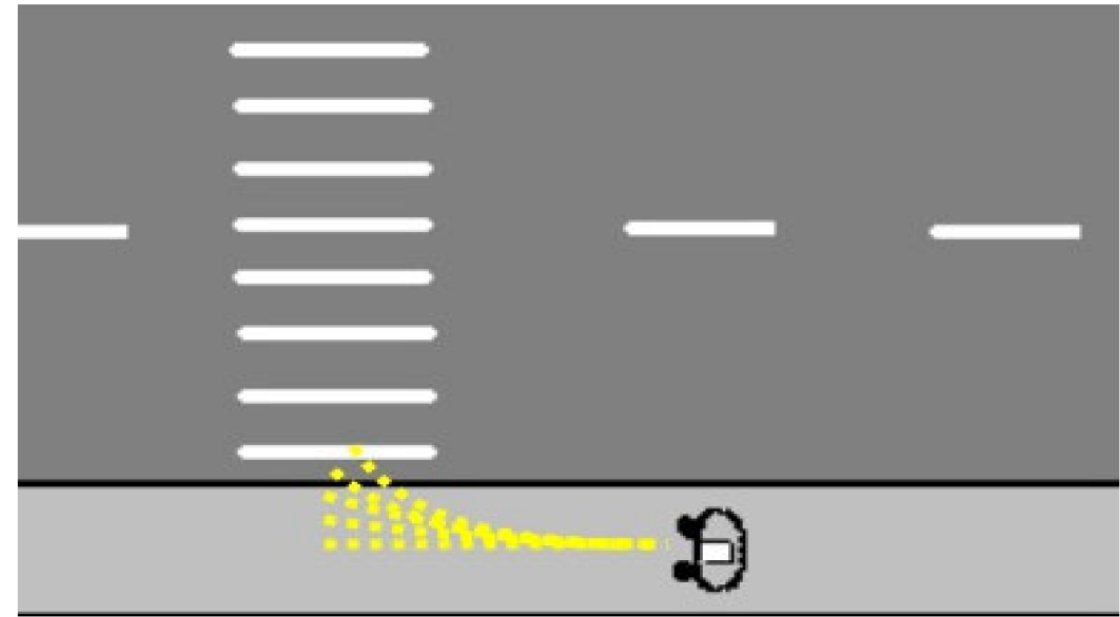
Modellierung der Umgebung von CPS:

- Modellierung mit KI/Deep Learning
 - Objektdetektion, -tracking, Bildsegmentierung, Greifpunkte
 - Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?
 - Spracherkennung
 - **Verhaltensmodellierung von Fußgängern, Fahrzeugen**
 - World Models
- Lernen in virtuellen Umgebungen (in einer Simulation)
 - 3D Simulation
 - Von der Simulation in die Realität (Sim2real)
- Statische Modelle der Umgebung (Karte)
 - SLAM – Simultaneous Localization and Mapping
 - Semantische Karte (Karte mit Objekten/Orten: Couch, TV, Küche, ...)

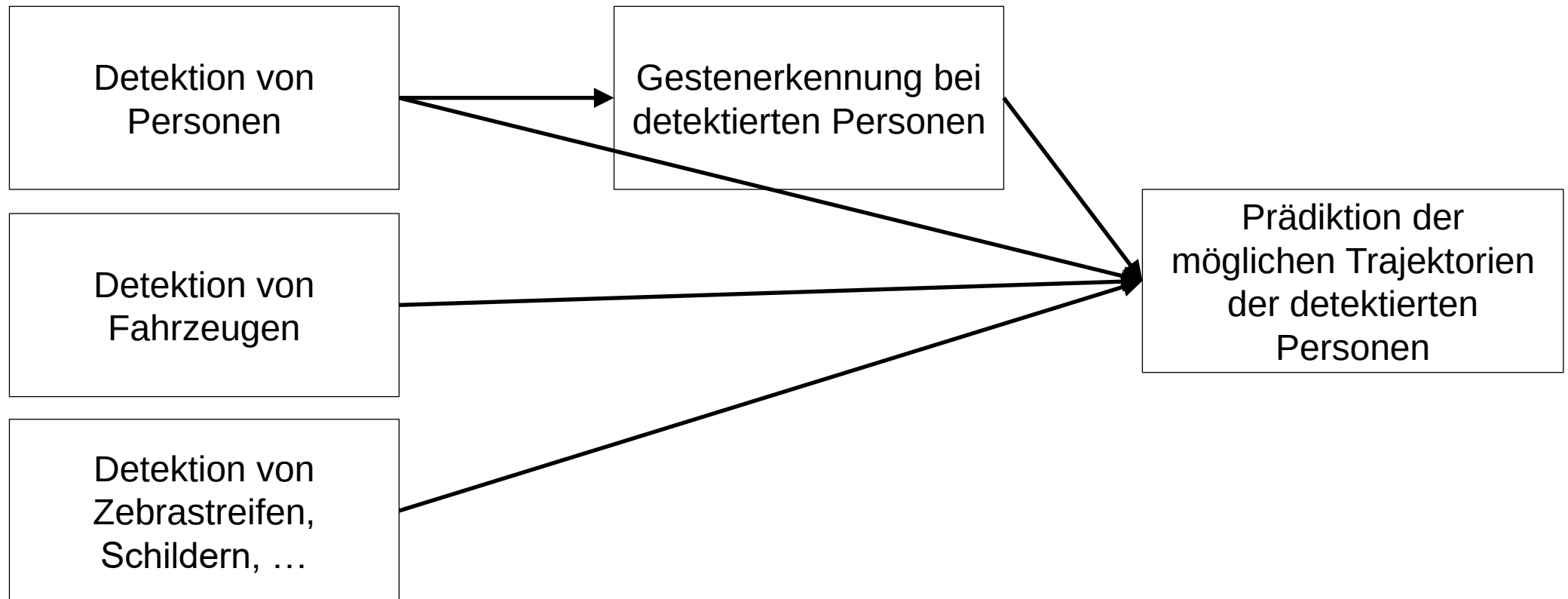
Verhaltensmodellierung von Fußgängern

Nutzung von Datenbasierten Modellen

- Lernen Wahrscheinlichkeiten für verschiedene Trajektorien
- Daten
 - Fußgänger und andere Objekte, Fahrzeuge, ...
 - Trajektorien der Fußgänger
 - Tätigkeiten der Fußgänger (Gesten)
 - Infrastruktur (Zebrastreifen, Schilder, ...)



Verhaltensmodellierung von Fußgängern



Verhaltensmodellierung von Fußgängern

Gestenerkennung – Human Pose Estimation – geht auch mit YOLOv8



Verhaltensmodellierung von Fußgängern

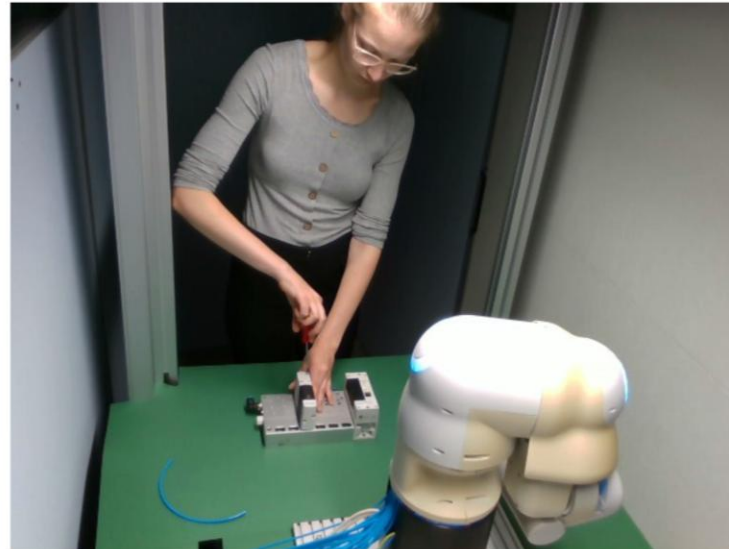
Gestenerkennung – Human Pose Estimation – geht auch mit YOLOv8



Tätigkeitserkennung von Personen



Bildquelle: <https://learnopencv.com/deep-learning-based-human-pose-estimation-using-opencv-cpp-python/>



Tätigkeitserkennung von Personen



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

<https://www.youtube.com/watch?v=3z3yiLdznrY>

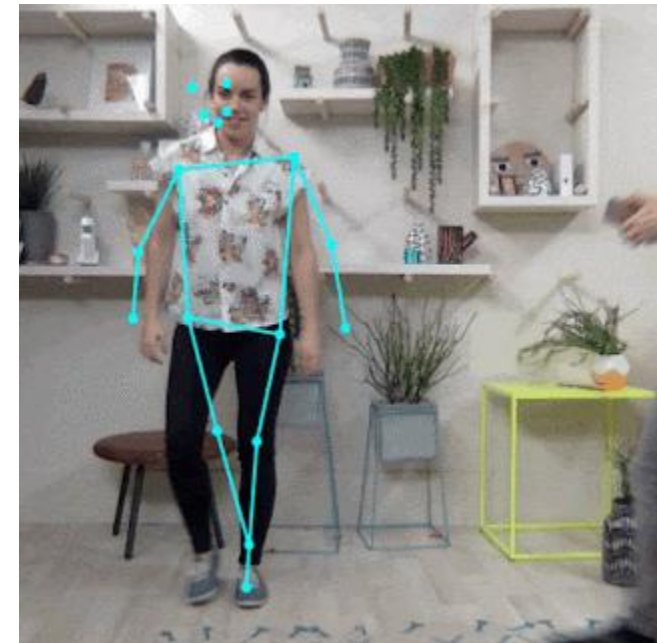
Angleraud, A., Ekrekli, A., Samarawickrama, K., Sharma, G., & Pieters, R. (2024). Sensor-based human-robot collaboration for industrial tasks. Robotics and Computer-Integrated Manufacturing, 86, 102663.

Tätigkeitserkennung von Personen

Aktivität

Gehen Sie zu und erlauben Sie dem Browser einen Zugriff auf Ihre Kamera:

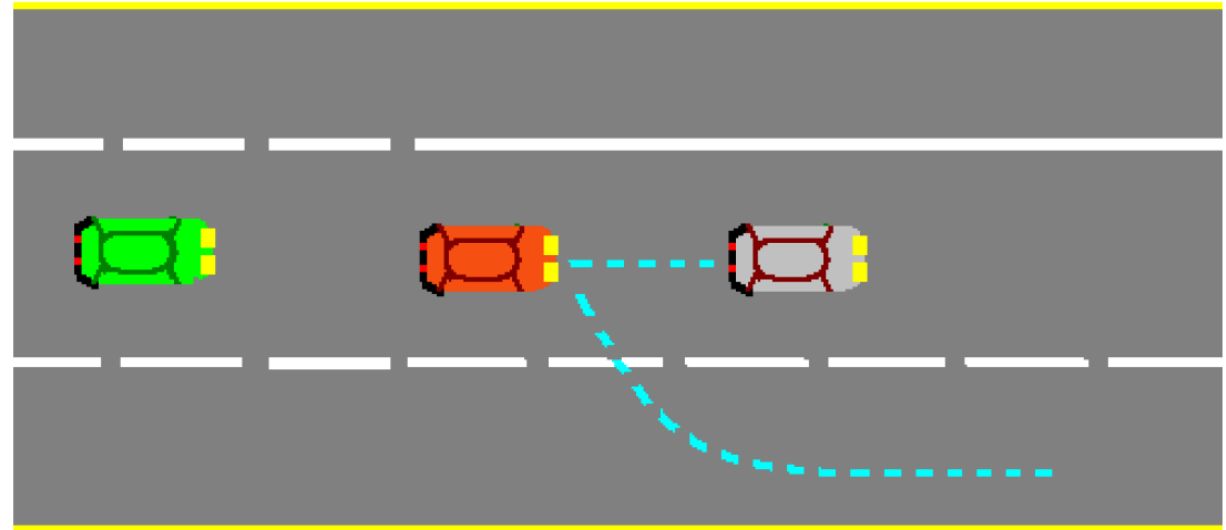
- <https://storage.googleapis.com/tfjs-models/demos/posenet/camera.html>



Verhaltensmodellierung von Fahrzeugen

Nutzung von Datenbasierten Modellen

- Lernen Wahrscheinlichkeiten für verschiedene Trajektorien
- Daten
 - Fahrzeuge und andere Objekte, Fußgänger, ...
 - Trajektorien der Fahrzeuge
 - Infrastruktur (Fahrbahnmarkierungen, Zebrastreifen, Schilder, ...)



Verhaltensmodellierung von Fahrradfahrern - Beispiel



Verhaltensmodellierung von Fahrradfahrern - Beispiel



- Objektdetektion/-tracking
 - Fahrradfahrer
 - Schilder
-
-
-

Verhaltensmodellierung von Fahrradfahrern - Beispiel



- Objektdetektion/-tracking
 - Fahrradfahrer
 - Schilder
- Gestenerkennung
 - Handzeichen und Drehung des Gesichts
- Fahrbahnmarkierung

Übung: Verhaltensmodellierung von Studierenden in der Mensa

- Ein mobiler Roboter sieht um 12:30 Uhr eine stehende/gehende Person in der Mensa
- Anhand welcher Daten kann der Roboter Vorhersagen für das Verhalten der Person in den nächsten Sekunden machen?

Übung: Verhaltensmodellierung von Studierenden in der Mensa

- Ein mobiler Roboter sieht um 12:30 Uhr eine stehende/gehende Person in der Mensa
- Anhand welcher Daten kann der Roboter Vorhersagen für das Verhalten der Person in den nächsten Sekunden machen?
- Hält die Person etwas in der Hand (bspw. Tablett)? Ist auf dem Tablett ein voller oder leerer Teller?
-
-

Übung: Verhaltensmodellierung von Studierenden in der Mensa

- Ein mobiler Roboter sieht um 12:30 Uhr eine stehende/gehende Person in der Mensa
- Anhand welcher Daten kann der Roboter Vorhersagen für das Verhalten der Person in den nächsten Sekunden machen?
- Hält die Person etwas in der Hand (bspw. Tablett)? Ist auf dem Tablett ein voller oder leerer Teller?
- Wo ist die Person in der Mensa? D.h., wo ist der Roboter und wo schaut der Roboter hin -> daraus Position der Person ermittelbar
-

Übung: Verhaltensmodellierung von Studierenden in der Mensa

- Ein mobiler Roboter sieht um 12:30 Uhr eine stehende/gehende Person in der Mensa
- Anhand welcher Daten kann der Roboter Vorhersagen für das Verhalten der Person in den nächsten Sekunden machen?
- Hält die Person etwas in der Hand (bspw. Tablett)? Ist auf dem Tablett ein voller oder leerer Teller?
- Wo ist die Person in der Mensa? D.h., wo ist der Roboter und wo schaut der Roboter hin -> daraus Position der Person ermittelbar
- Wo schaut die Person hin? Macht sie Gesten? Ist sie alleine oder zusammen mit anderen Personen?

Inhalt

Modellierung der Umgebung von CPS:

- Modellierung mit KI/Deep Learning
 - Objektdetektion, -tracking, Bildsegmentierung, Greifpunkte
 - Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?
 - Spracherkennung
 - Verhaltensmodellierung von Fußgängern, Fahrzeugen
 - **World Models**
- Lernen in virtuellen Umgebungen (in einer Simulation)
 - 3D Simulation
 - Von der Simulation in die Realität (Sim2real)
- Statische Modelle der Umgebung (Karte)
 - Semantische Karte (Karte mit Objekten/Orten: Couch, TV, Küche, ...)
 - SLAM – Simultaneous Localization and Mapping

World Models: Der „interne Simulator“ für CPS

- Was ist ein World Model?
 - Ein generatives Modell, das die Übergangsdynamik der Umgebung erlernt: Es prognostiziert zukünftige Umweltzustände auf Basis aktueller Beobachtungen und geplanter Aktionen.
 - Es fungiert als lernfähiger digitaler Zwilling bzw. mentaler Simulator der physikalischen Welt.
- - 1.
 - 2.
 - 3.

World Models: Der „interne Simulator“ für CPS

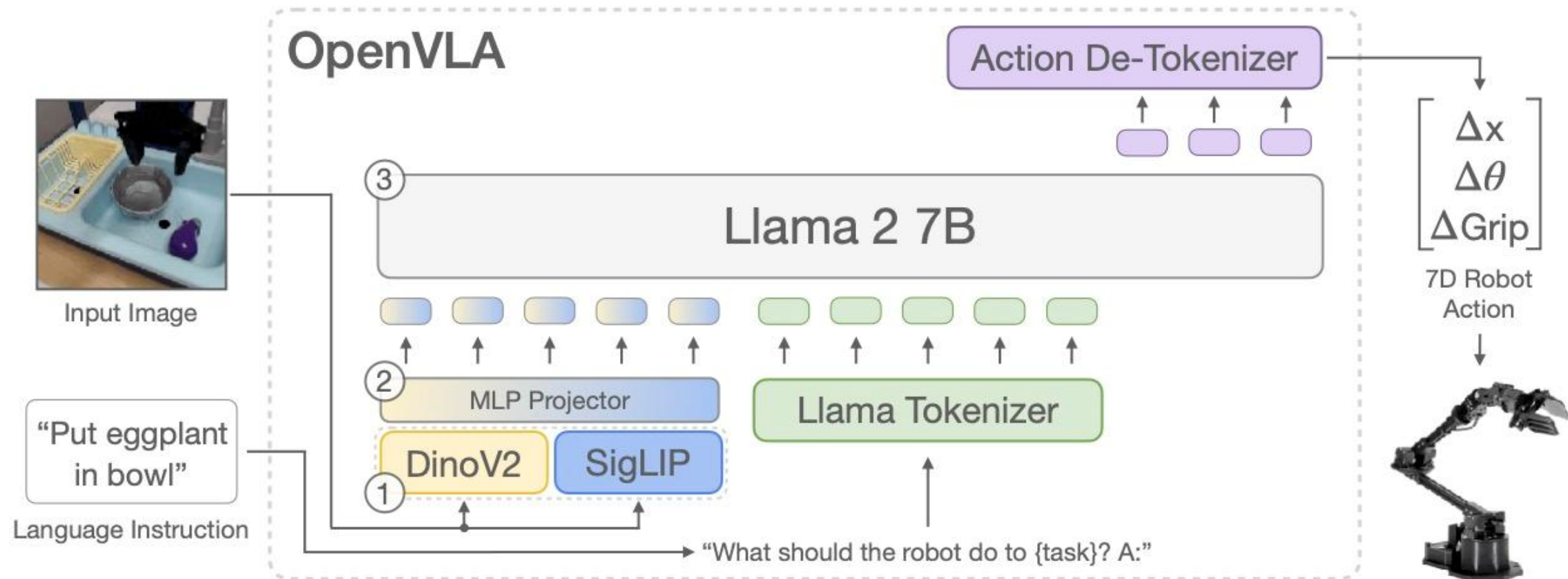
- Was ist ein World Model?
 - Ein generatives Modell, das die Übergangsdynamik der Umgebung erlernt: Es prognostiziert zukünftige Umweltzustände auf Basis aktueller Beobachtungen und geplanter Aktionen.
 - Es fungiert als lernfähiger digitaler Zwilling bzw. mentaler Simulator der physikalischen Welt.
- Das Versprechen:
 1. Plausibel: Behält geometrische Strukturen und physikalische Objektpermanenz über die Zeit stabil bei
 2. Steuerbar: Reagiert kausal und differenziert auf konkrete Steuereingriffe des Roboters
 - 3.

World Models: Der „interne Simulator“ für CPS

- Was ist ein World Model?
 - Ein generatives Modell, das die Übergangsdynamik der Umgebung erlernt: Es prognostiziert zukünftige Umweltzustände auf Basis aktueller Beobachtungen und geplanter Aktionen.
 - Es fungiert als lernfähiger digitaler Zwilling bzw. mentaler Simulator der physikalischen Welt.
- Das Versprechen:
 1. Plausibel: Behält geometrische Strukturen und physikalische Objektpermanenz über die Zeit stabil bei
 2. Steuerbar: Reagiert kausal und differenziert auf konkrete Steuereingriffe des Roboters
 3. Handlungswirksam: Führt zu messbar besseren Entscheidungen, sicherer Planung und Fehlerkorrektur zur Laufzeit

World Models

Evolution der Robotersteuerung: Vision-Language-Action Modelle



World Models

Evolution der Robotersteuerung: Von reaktiven Reflexen (VLAs)...

- Das Prinzip von Vision-Language-Action (VLA):
 - Internet-vortrainierte Vision-Language-Modelle (VLMs) werden mit Demonstrationsdaten feingetunt, um Sprachbefehle und Kamerabilder direkt auf Roboteraktionen abzubilden
- Der große Durchbruch:
 - Hohe semantische Generalisierung: Roboter verstehen neuartige Objekte und sprachliche Anweisungen ohne aufgabenspezifische Neuprogrammierung.
- -
 -

World Models

Evolution der Robotersteuerung: Von reaktiven Reflexen (VLAs)...

- Das Prinzip von Vision-Language-Action (VLA):
 - Internet-vortrainierte Vision-Language-Modelle (VLMs) werden mit Demonstrationsdaten feingetunt, um Sprachbefehle und Kamerabilder direkt auf Roboteraktionen abzubilden
- Der große Durchbruch:
 - Hohe semantische Generalisierung: Roboter verstehen neuartige Objekte und sprachliche Anweisungen ohne aufgabenspezifische Neuprogrammierung.
- Warum VLAs für CPS an Grenzen stoßen:
 - Rein reaktive Steuerung: Ein Standard-VLA besitzt keinen Mechanismus, um die zeitlichen oder physikalischen Konsequenzen seiner Handlungen vorab zu modellieren.
 -

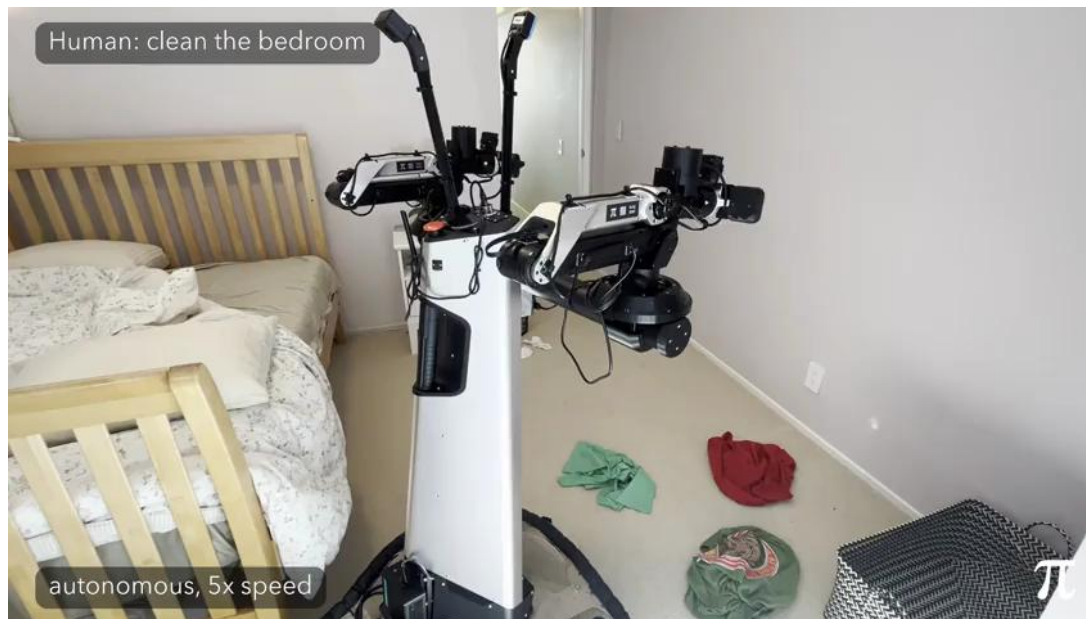
World Models

Evolution der Robotersteuerung: Von reaktiven Reflexen (VLAs)...

- Das Prinzip von Vision-Language-Action (VLA):
 - Internet-vortrainierte Vision-Language-Modelle (VLMs) werden mit Demonstrationsdaten feingetunt, um Sprachbefehle und Kamerabilder direkt auf Roboteraktionen abzubilden
- Der große Durchbruch:
 - Hohe semantische Generalisierung: Roboter verstehen neuartige Objekte und sprachliche Anweisungen ohne aufgabenspezifische Neuprogrammierung.
- Warum VLAs für CPS an Grenzen stoßen:
 - Rein reaktive Steuerung: Ein Standard-VLA besitzt keinen Mechanismus, um die zeitlichen oder physikalischen Konsequenzen seiner Handlungen vorab zu modellieren.
 - Keine kausale Vorausschau: Bei Verdeckungen (z. B. Hand verdeckt Greifobjekt) oder unvorhergesehenen Störungen fehlt dem Modell das Verständnis dafür, wie sich die Umwelt physikalisch weiterentwickeln sollte

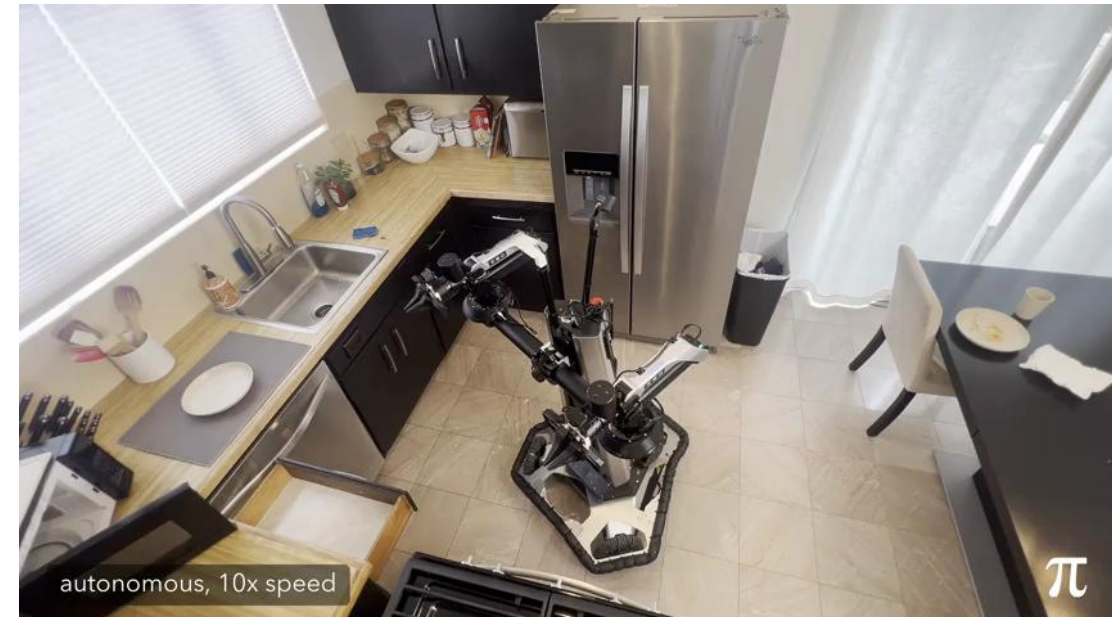
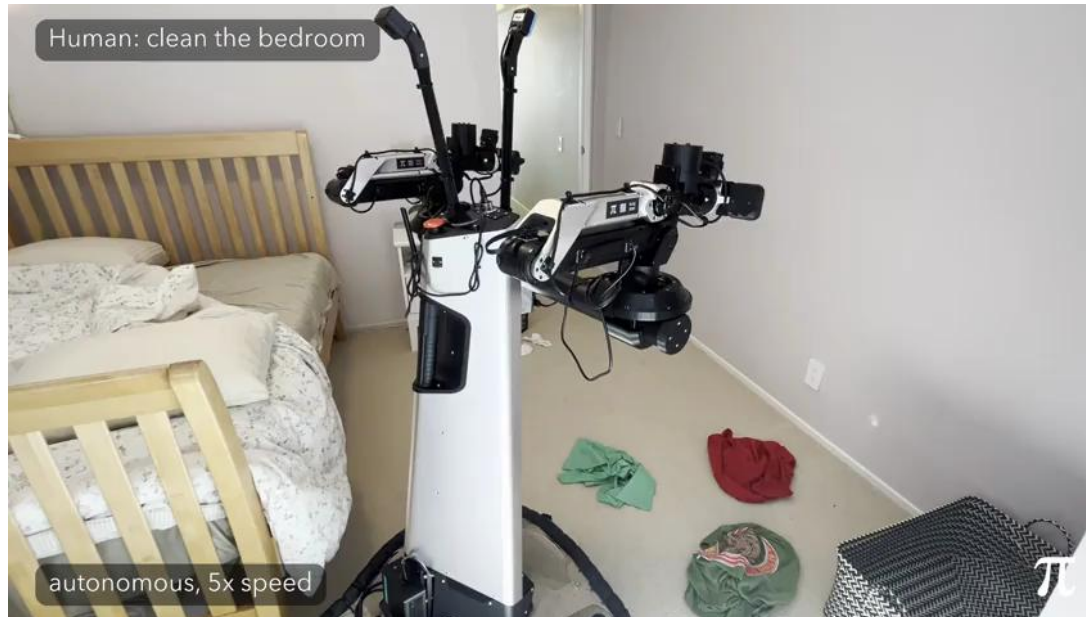
Physical Intelligence: $\pi 0.5$: a VLA with Open-World Generalization

Räume das Schlafzimmer auf



Physical Intelligence: π 0.5: a VLA with Open-World Generalization

Räume des Schlafzimmers auf



World Models

...zu World Action Models: Vorausschau und Aktion vereint

- Der Paradigmenwechsel zu World Action Models:
 - Kopplung von Zustandsvorhersage und Aktionsauswahl
 - Die antizipierte Zukunft bleibt direkt im Aktionspfad: Sie dient unmittelbar dazu, Aktionen zu generieren, zu bewerten oder zu filtern
- -
 -

World Models

...zu World Action Models: Vorausschau und Aktion vereint

- Der Paradigmenwechsel zu World Action Models:
 - Kopplung von Zustandsvorhersage und Aktionsauswahl
 - Die antizipierte Zukunft bleibt direkt im Aktionspfad: Sie dient unmittelbar dazu, Aktionen zu generieren, zu bewerten oder zu filtern
- Der Mehrwert für robuste Cyber-Physische Systeme:
 - Laufzeit-Verifikation: Weicht die echte Sensorbeobachtung von der prognostizierten Folgebeobachtung ab (z. B. Objekt rutscht ab), erkennt das System den Fehler sofort und korrigiert ihn.
 -

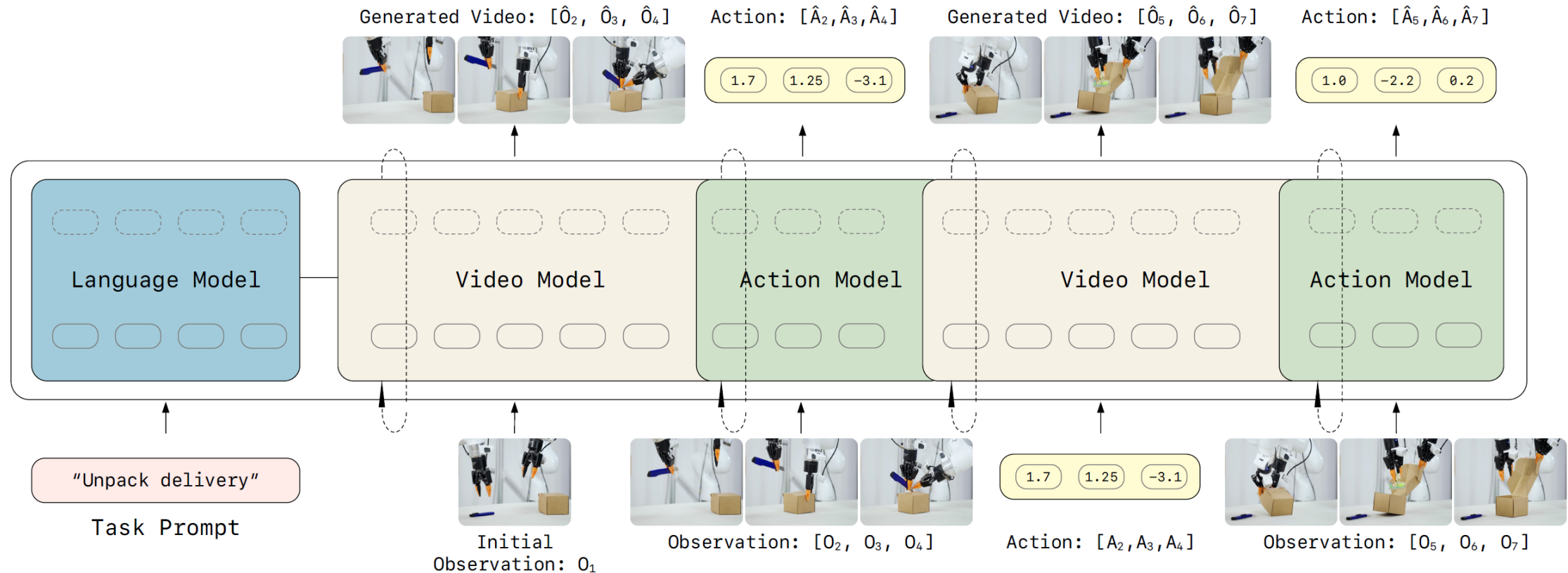
World Models

...zu World Action Models: Vorausschau und Aktion vereint

- Der Paradigmenwechsel zu World Action Models:
 - Kopplung von Zustandsvorhersage und Aktionsauswahl
 - Die antizipierte Zukunft bleibt direkt im Aktionspfad: Sie dient unmittelbar dazu, Aktionen zu generieren, zu bewerten oder zu filtern
- Der Mehrwert für robuste Cyber-Physische Systeme:
 - Laufzeit-Verifikation: Weicht die echte Sensorbeobachtung von der prognostizierten Folgebeobachtung ab (z. B. Objekt rutscht ab), erkennt das System den Fehler sofort und korrigiert ihn.
 - Sichere Imagination: CPS können neue oder riskante Interaktionen in der internen Simulation durchspielen, ohne reale Hardware zu beschädigen

World Models

...zu World Action Models: Vorausschau und Aktion vereint



World Models

...zu World Action Models: Vorausschau und Aktion vereint



Inhalt

Modellierung der Umgebung von CPS:

- Modellierung mit KI/Deep Learning
 - Objektdetektion, -tracking, Bildsegmentierung, Greifpunkte
 - Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?
 - Spracherkennung
 - Verhaltensmodellierung von Fußgängern, Fahrzeugen
 - World Models
- **Lernen in virtuellen Umgebungen (in einer Simulation)**
 - **3D Simulation**
 - **Von der Simulation in die Realität (Sim2real)**
- Statische Modelle der Umgebung (Karte)
 - SLAM – Simultaneous Localization and Mapping
 - Semantische Karte (Karte mit Objekten/Orten: Couch, TV, Küche, ...)

Lernen durch Simulation

- Training von Deep Learning Modellen benötigt sehr viele Daten
- Erhebung ist teuer, zeitintensiv, aufwändig, ...



Lernen durch Simulation

- Training von Deep Learning Modellen benötigt sehr viele Daten
- Erhebung ist teuer, zeitintensiv, aufwändig, ...
- Lösung:
 - Erstellen der Daten in einer fotorealistischen Simulationsumgebung
 - Weiterer Vorteil: Daten sind bereits gelabelt
- Training des Deep Learning Modells mit den simulierten Daten
- Fine-Tuning des trainierten Modells mit **wenigen** realen Daten

Was ist eine Simulation?

- Eine Simulation ist eine virtuelle Welt, die eine Teilmenge von Merkmalen der realen Welt enthält.
- Im Zusammenhang mit der Robotik sind Simulationen virtuelle Welten, die es Entwicklern ermöglichen, produktiver zu arbeiten.
- Simulationen sind modernen Videospielen insofern sehr ähnlich, als sie realistische Physiksimulationen, eine Vorstellung von Zeit und grafische Darstellungen der realen Welt enthalten.

3D Simulation

CARLA, Open Source, <https://carla.readthedocs.io/en/latest/>

Simulation von

- Sensoren
 - Kamera, Lidar, GPS, ...
- Physik (Trägheit, Reibung, ...)
- Wetter, Licht, ...

3D Simulation

CARLA, Open Source, <https://carla.readthedocs.io/en/latest/>



Simulation von

- Sensoren
 - Kamera, Lidar, GPS, ...
- Physik (Trägheit, Reibung, ...)
- Wetter, Licht, ...

3D Simulation

Objektdetektion in CARLA



3D Simulation

Nvidia DRIVE Sim – We are not accepting new partners currently



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

<https://developer.nvidia.com/drive/simulation>

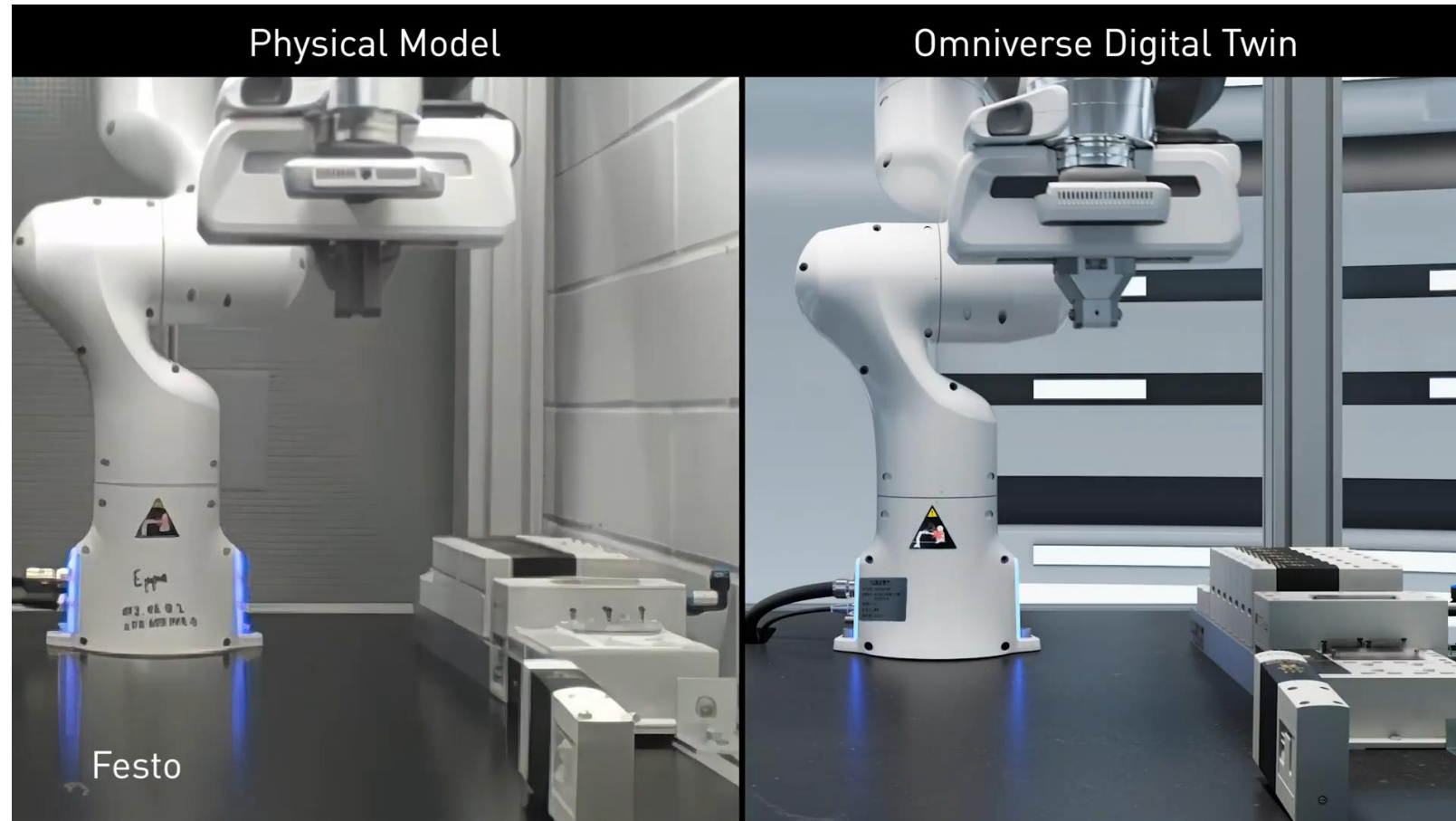
Technology
Arts Sciences
TH Köln

3D Simulation

Nvidia Isaac Sim (kostenlos, Datenschutz problematisch)

3D Simulation

Nvidia Isaac Sim (kostenlos, Datenschutz problematisch)



3D Simulation

Nvidia Omniverse & Amazon

3D Simulation

Nvidia Omniverse & Amazon



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

<https://www.youtube.com/watch?v=-VQLqs6s9y0>

3D Simulation

Nvidia Omniverse & Amazon



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

https://www.youtube.com/watch?v=LUnZXBL_lqA

Technology
Arts Sciences
TH Köln

3D Simulation

Nvidia Omniverse & Siemens



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

<https://www.youtube.com/watch?v=t6ppwWZUSEc>

Technology
Arts Sciences
TH Köln

3D Simulation

Gazebo, Open Source



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

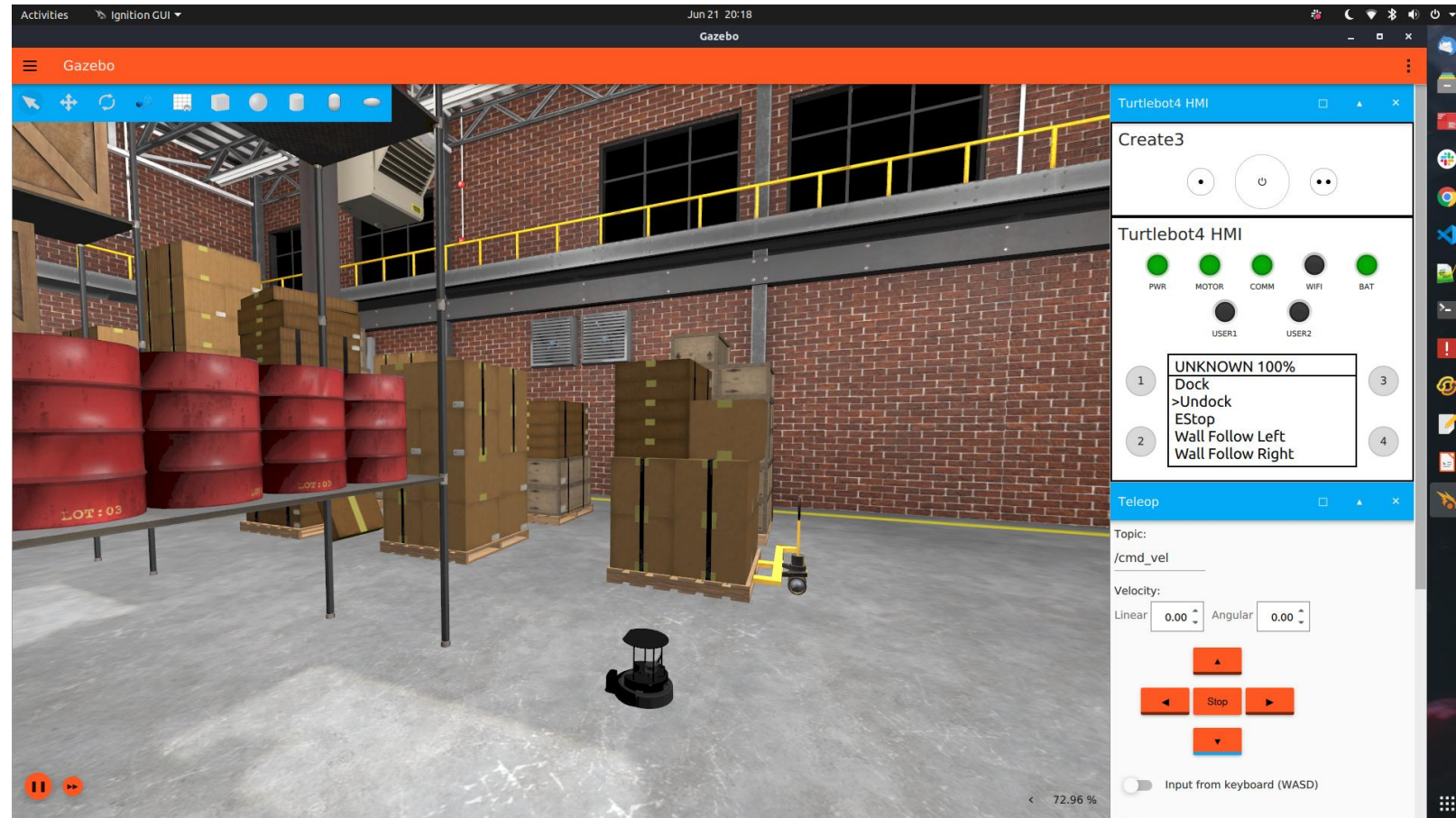
<https://gazebosim.org/home>

3D Simulation

Gazebo, Open Source



<https://raw.githubusercontent.com/turtlebot/TurtleBot4Lessons/main/media/tb4.png>



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

<https://github.com/turtlebot/TurtleBot4Lessons/blob/main/units/Unit01-ROS-2-Basics/L02-Gazebo-World/U01-L02-Empty-World.md>

3D Simulation

Habitat-Lab, Open Source

Simulation von Innenräumen

- Navigation („Gehe in die Küche“)
- Anweisungen folgen („Drehe dich nach rechts, gehe durch die Tür, ...“)
- Beantwortung von Fragen („Welche Farbe hat die Haustür?“)

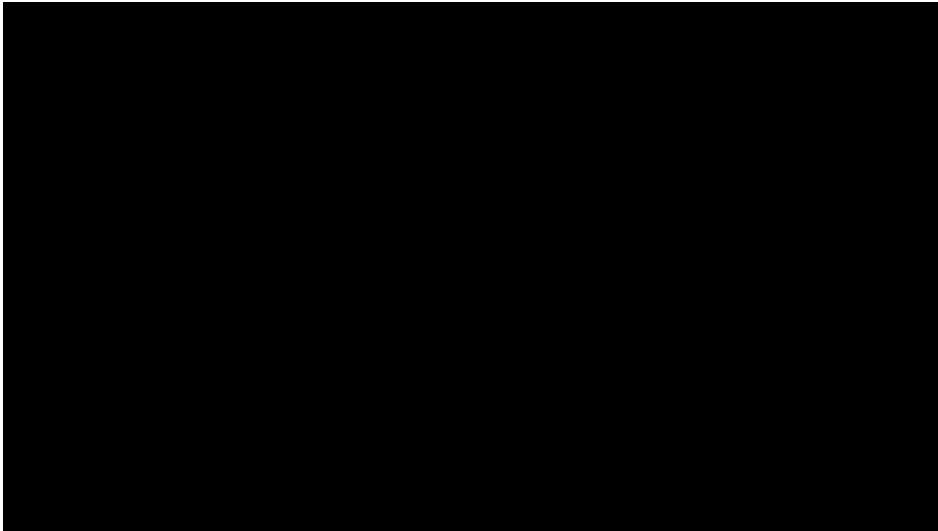


3D Simulation

Habitat-Lab, Open Source

Simulation von Innenräumen

- Navigation („Gehe in die Küche“)
- Anweisungen folgen („Drehe dich nach rechts, gehe durch die Tür, ...“)
- Beantwortung von Fragen („Welche Farbe hat die Haustür?“)



3D Simulation

ThreeDWorld, Open Source

“TDW is a 3D virtual world simulation platform, utilizing state-of-the-art video game engine technology”

<https://threedworld.org/>



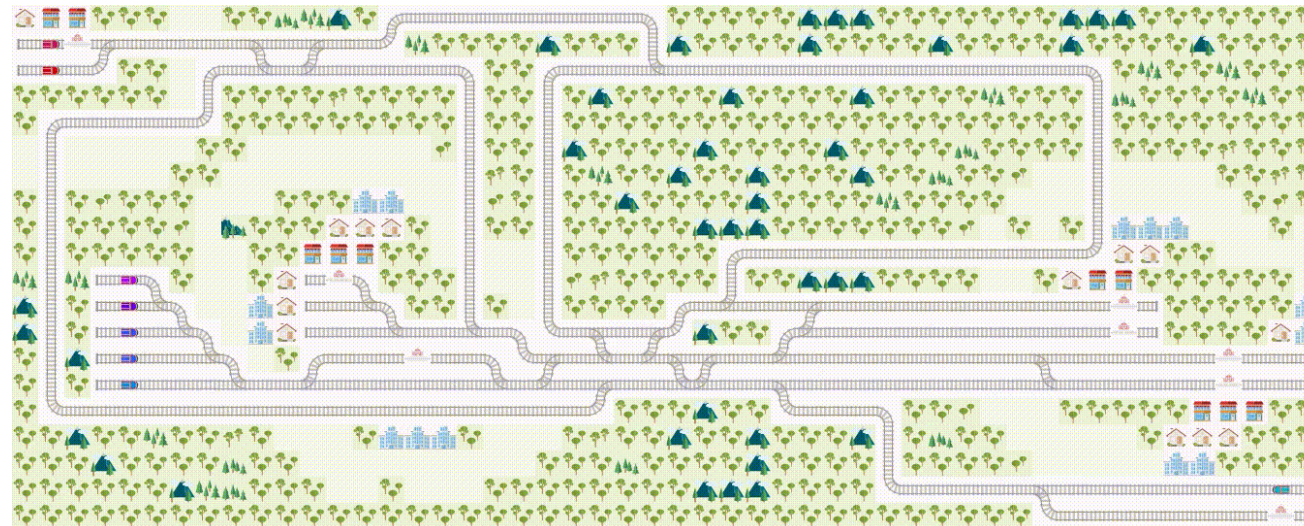
Weitere Simulationen

- EnergyPlus, „EnergyPlus™ is a whole building energy simulation program that engineers, architects, and researchers use to model both energy consumption - for heating, cooling, ventilation, lighting and plug and process loads - and water use in buildings.“, Quelle: <https://energyplus.net/>

■

Weitere Simulationen

- EnergyPlus, „EnergyPlus™ is a whole building energy simulation program that engineers, architects, and researchers use to model both energy consumption - for heating, cooling, ventilation, lighting and plug and process loads - and water use in buildings.“, Quelle: <https://energyplus.net/>
- Flatland, How to efficiently manage dense traffic on complex railway networks?, <https://flatland.aicrowd.com/intro.html>



Von der Simulation in die Realität

Sim2Real, sim-to-real

Sim2real bezieht sich auf den Prozess der Übertragung einer in einer simulierten Umgebung erlernten Strategie oder eines Modells auf eine reale Umgebung.

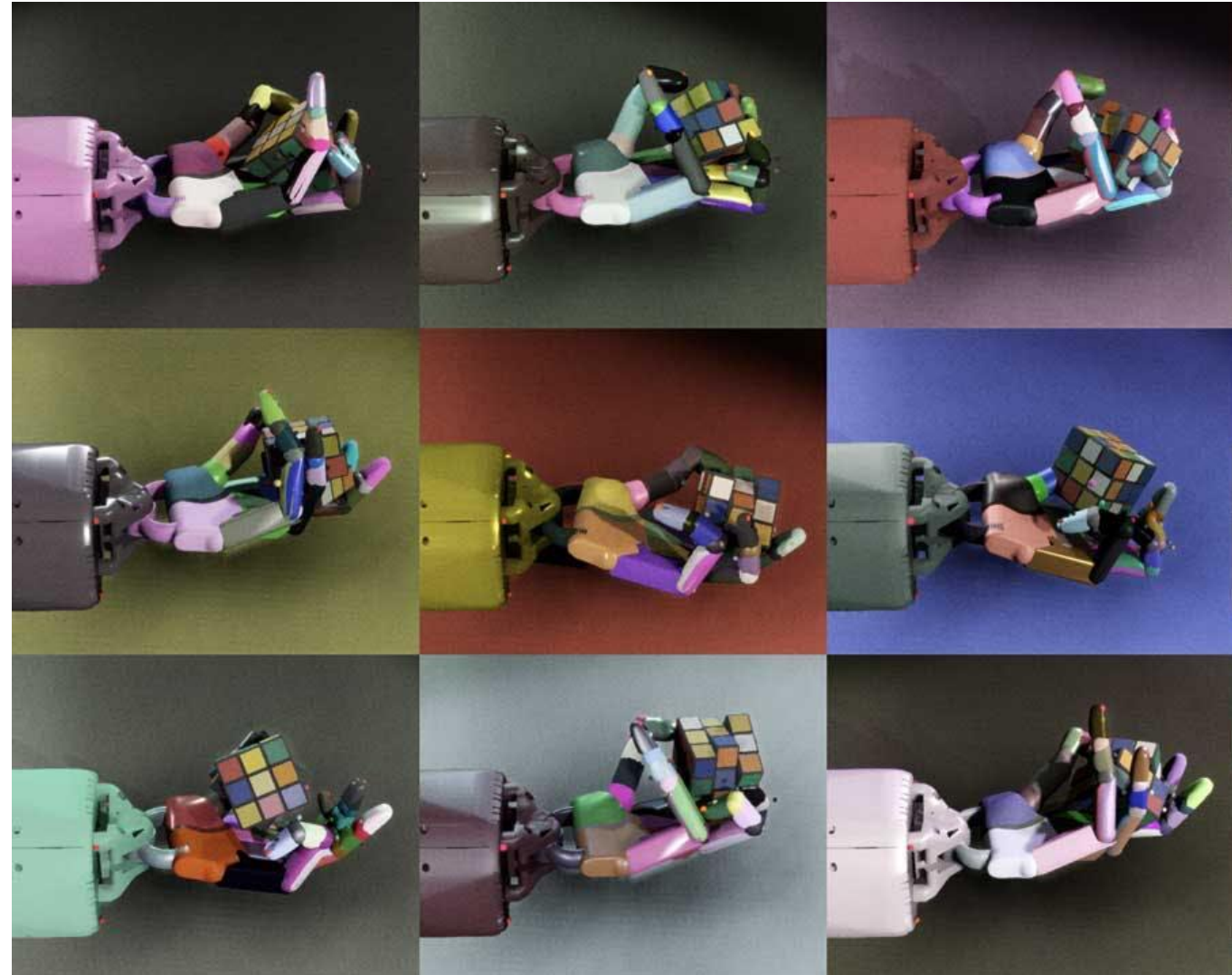
- Schließen des „reality gaps“ (Lücke zwischen Simulation und Realität)

Simulation → Realität

Sim2Real, sim-to-real

Domain Randomization:

- Anwendung von zufälligen Transformationen auf ein Simulationsmodell, um die Generalisierung auf unbekannte Umgebungen zu verbessern.
- Training einer Roboterhand, um den Zauberwürfel (Rubik's Cube) zu lösen
-

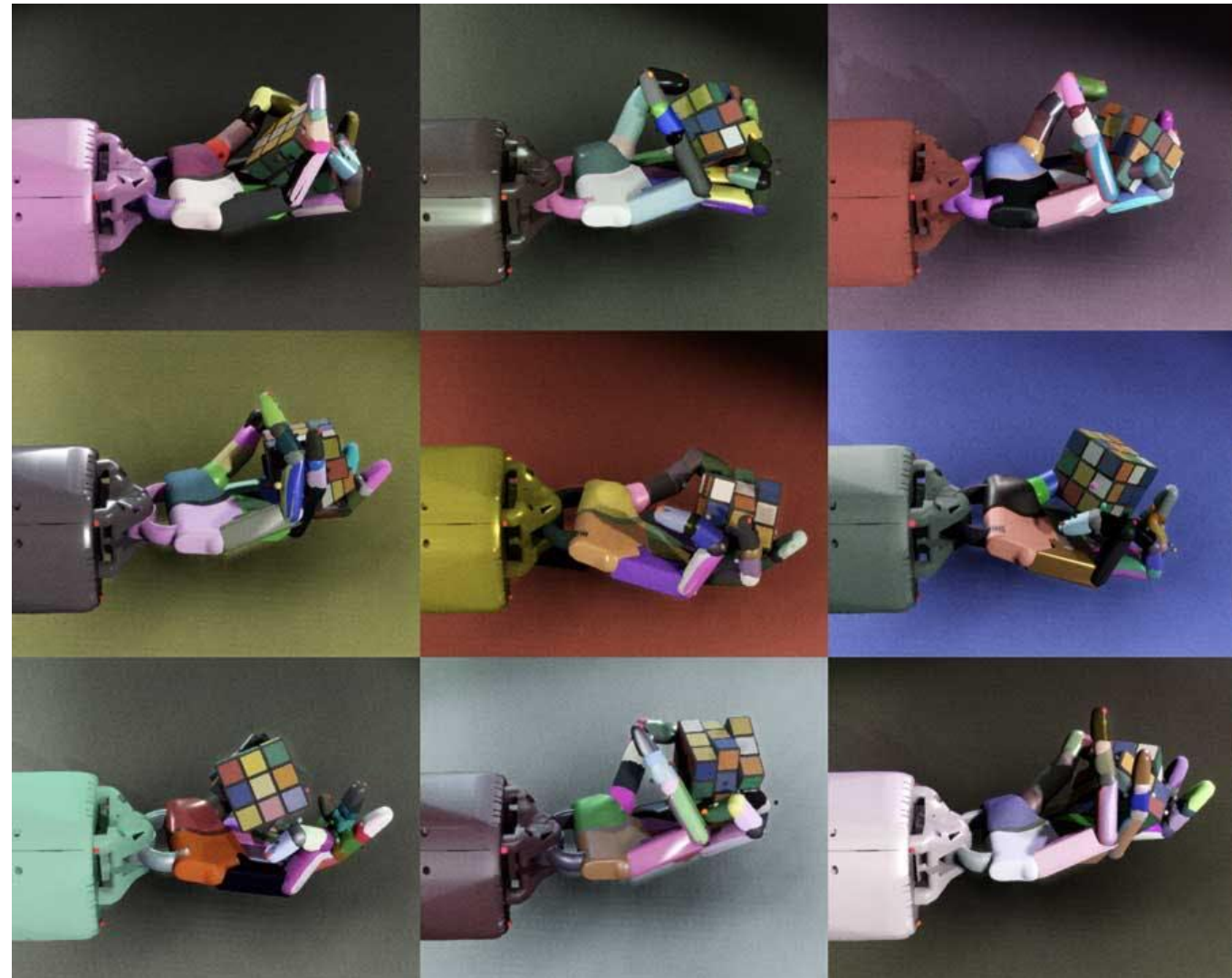


Simulation → Realität

Sim2Real, sim-to-real

Domain Randomization:

- Anwendung von zufälligen Transformationen auf ein Simulationsmodell, um die Generalisierung auf unbekannte Umgebungen zu verbessern.
- Training einer Roboterhand, um den Zauberwürfel (Rubik's Cube) zu lösen
- Simulation verschieden aussehender Roboterhände, damit KI lernt, dass Aussehen der Hand keine Rolle spielt

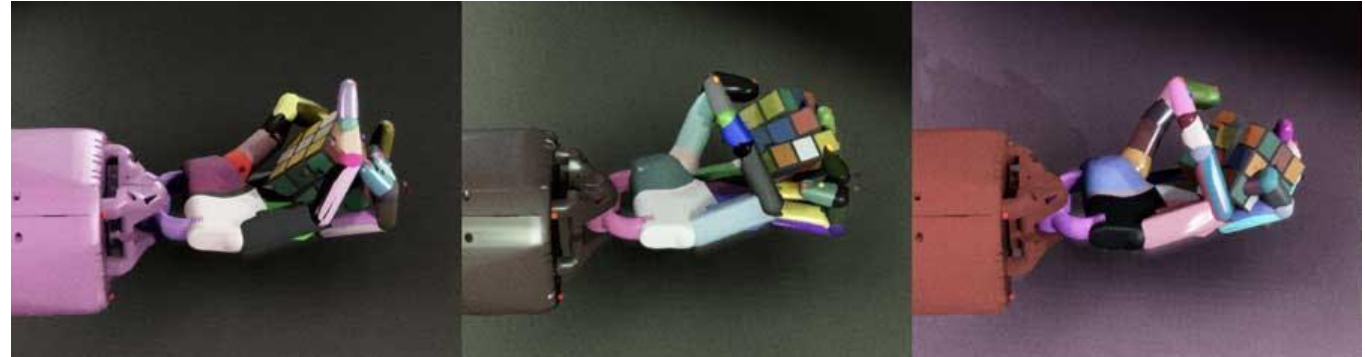


Simulation → Realität

Sim2Real, sim-to-real

Domain Randomization:

- Randomisierung von visuellen Merkmalen, aber auch von physikalischen Parametern (Reibung, Masse, Größe von Objekten, ...)



Simulation → Realität

Sim2Real, sim-to-real

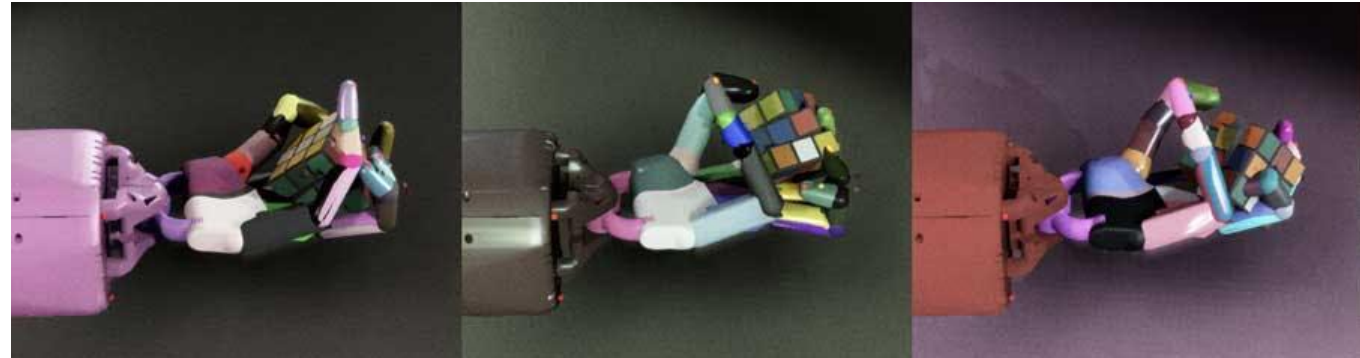
Domain Randomization:

- Randomisierung von visuellen Merkmalen, aber auch von physikalischen Parametern (Reibung, Masse, Größe von Objekten, ...)

- Ziel:

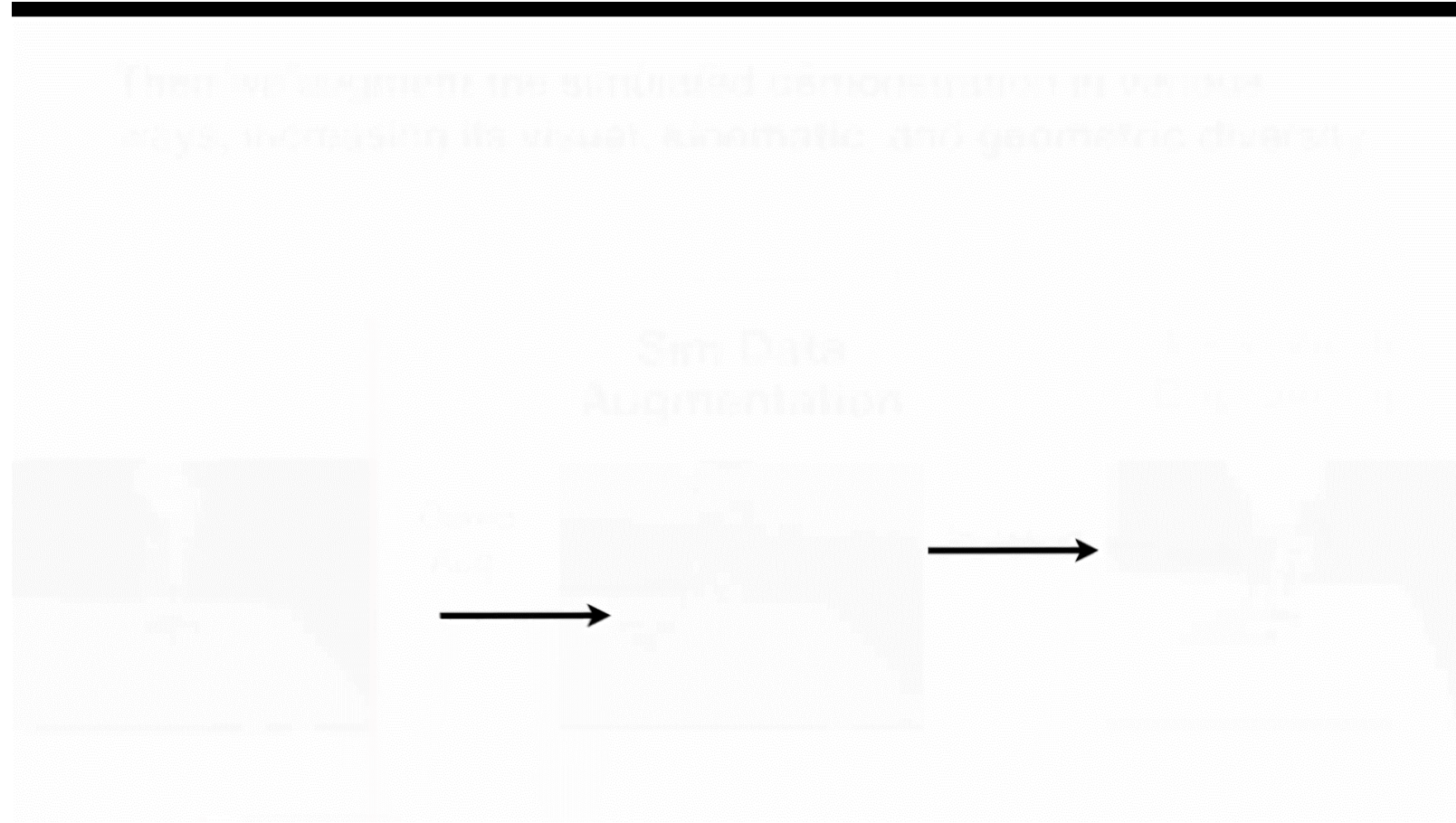
„Die Hauptidee hinter diesen Methoden ist das Lernen in einer großen Anzahl von simulierten Umgebungen mit dem Ziel, ein robustes Modell oder eine robuste Strategie zu erlernen, die die physische Welt als eine weitere Variation "sieht" und ohne jegliche Anpassung übertragen wird.“

[Quelle: K. Dimitropoulos et al., A Brief Survey of Sim2Real Methods for Robot Learning, 2022]



Simulation → Realität

Sim2Real, sim-to-real



Simulation → Realität

Sim2Real, sim-to-real

Werden Modelle/Strategien immer besser in der realen Welt funktionieren, wenn die Simulationen immer besser/realistischer werden?

■

■

■

Simulation → Realität

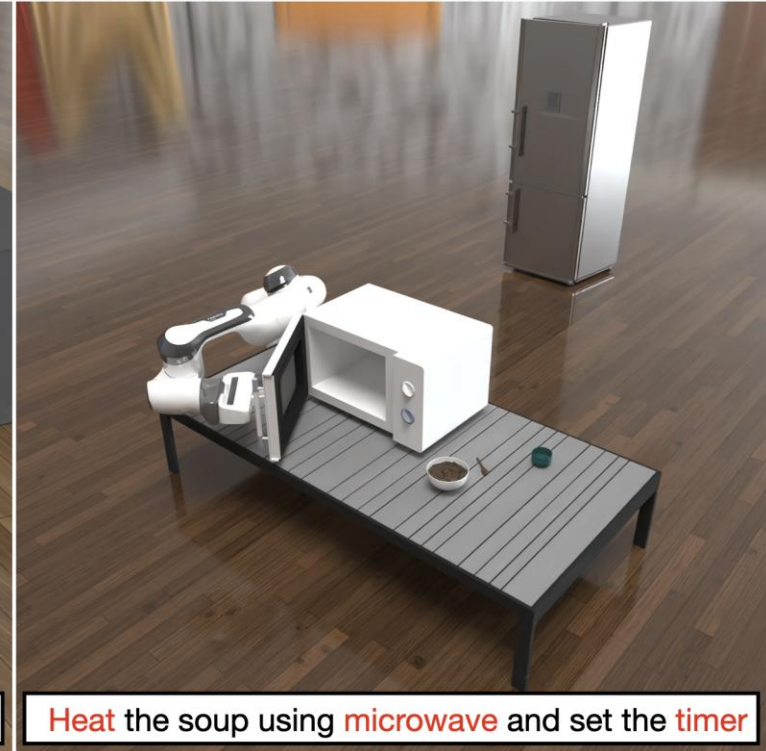
Sim2Real, sim-to-real

Werden Modelle/Strategien immer besser in der realen Welt funktionieren, wenn die Simulationen immer besser/realistischer werden?

- Scheinbar nicht
- “Our results show that, contrary to expectation, adding fidelity does not help with learning; performance is poor due to slow simulation speed (preventing large-scale learning) and overfitting to inaccuracies in simulation physics.” aus: J. Truong et al., Rethinking Sim2Real: Lower Fidelity Simulation Leads to Higher Sim2Real Transfer in Navigation, 2022
- Übersetzung: “Unsere Ergebnisse zeigen, dass die Erhöhung der Genauigkeit entgegen den Erwartungen nicht zum Lernen beiträgt; die Leistung ist aufgrund der langsamen Simulationsgeschwindigkeit (die ein Lernen in großem Maßstab verhindert) und der Überanpassung an Ungenauigkeiten in der Simulationsphysik schlecht.”

Simulation → Realität

Sim2Real, sim-to-real



Simulation → Realität

Sim2Real, sim-to-real

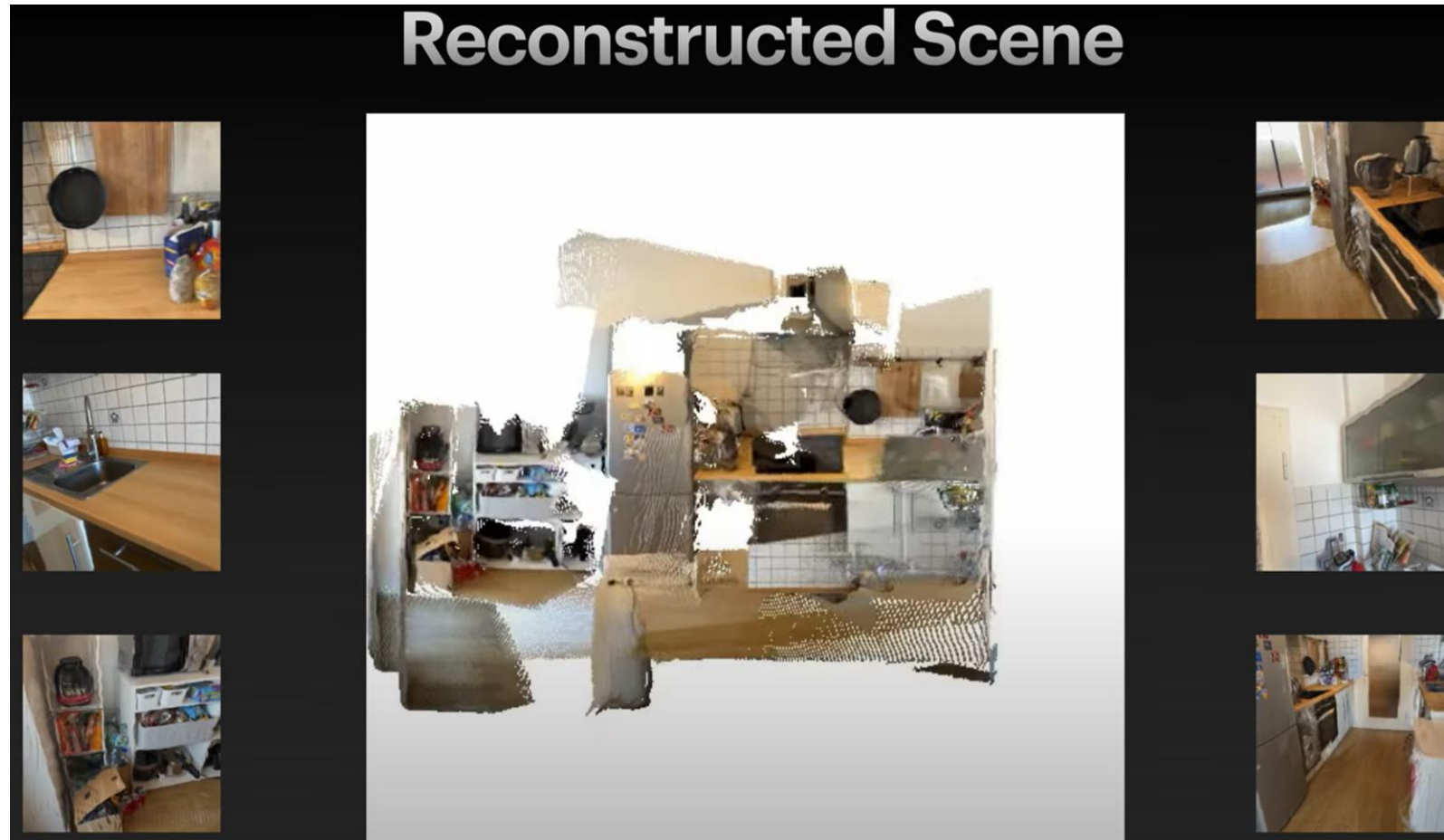


Simulation → Realität

Sim2Real, sim-to-real

Erstellung von Simulationsmodellen aus einer realen Umgebung

Gaussian Splatting



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

Quelle: <https://mv-dust3rp.github.io/>
MV-DUST3R+: Single-Stage Scene Reconstruction
from Sparse Views In 2 Seconds

Erstellung von Simulationsmodellen aus einer realen Umgebung

Gaussian Splatting



<https://superspl.at/editor>

13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

Quelle: N. Struckmeyer: Entwicklung eines Partikelfilters zur Lokalisierung mobiler Roboter mithilfe von 3D-Gaussian-Splatting, Bachelorthesis, 2026

Erstellung von Simulationsmodellen aus einer realen Umgebung

Gaussian Splatting



<https://superspl.at/editor>



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

Quelle: N. Struckmeyer: Entwicklung eines Partikelfilters zur Lokalisierung mobiler Roboter mithilfe von 3D-Gaussian-Splatting, Bachelorthesis, 2026

Übung – Erstellen eines Modells der Umgebung

- Entwicklung eines humanoiden Roboters, der beim Golf dem Golfspieler immer den richtigen Schläger anreicht.

-

-

-

Übung – Erstellen eines Modells der Umgebung

- Entwicklung eines humanoiden Roboters, der beim Golf dem Golfspieler immer den richtigen Schläger anreicht.
- Was ist die Umgebung?
 - Golfplatz, Schläger, Golfball, Spieler, ...
 -
 -
 -
 -
 -
-

Übung – Erstellen eines Modells der Umgebung

- Entwicklung eines humanoiden Roboters, der beim Golf dem Golfspieler immer den richtigen Schläger anreicht.
- Was ist die Umgebung?
 - Golfplatz, Schläger, Golfball, Spieler, ...
 -
 -
 -
 -
 -
-

Übung – Erstellen eines Modells der Umgebung

- Entwicklung eines humanoiden Roboters, der beim Golf dem Golfspieler immer den richtigen Schläger anreicht.
- Was ist die Umgebung?
 - Golfplatz, Schläger, Golfball, Spieler, ...
- Wie können wir diese modellieren?
 - Objekt Detektion, Tracking, Pose Estimation und Spracherkennung für Spieler
 - Objekt Detektion für Schläger, Fahne am Loch und Ball
 - Bildsegmentierung zur Modellierung der aktuellen Umgebung und Entscheidung welcher Golfschläger der am Besten geeignete ist
 - Spracherkennung für Keyword Spotting (Schlägernamen, ...)

■

Übung – Erstellen eines Modells der Umgebung

- Entwicklung eines humanoiden Roboters, der beim Golf dem Golfspieler immer den richtigen Schläger anreicht.
- Was ist die Umgebung?
 - Golfplatz, Schläger, Golfball, Spieler, ...
- Wie können wir diese modellieren?
 - Objekt Detektion, Tracking, Pose Estimation und Spracherkennung für Spieler
 - Objekt Detektion für Schläger, Fahne am Loch und Ball
 - Bildsegmentierung zur Modellierung der aktuellen Umgebung und Entscheidung welcher Golfschläger der am Besten geeignete ist
 - Spracherkennung für Keyword Spotting (Schlägernamen, ...)

■

Übung – Erstellen eines Modells der Umgebung

- Entwicklung eines humanoiden Roboters, der beim Golf dem Golfspieler immer den richtigen Schläger anreicht.
- Was ist die Umgebung?
 - Golfplatz, Schläger, Golfball, Spieler, ...
- Wie können wir diese modellieren?
 - Objekt Detektion, Tracking, Pose Estimation und Spracherkennung für Spieler
 - Objekt Detektion für Schläger, Fahne am Loch und Ball
 - Bildsegmentierung zur Modellierung der aktuellen Umgebung und Entscheidung welcher Golfschläger der am Besten geeignete ist
 - Spracherkennung für Keyword Spotting (Schlägernamen, ...)

- Trainingsdaten evtl. mit einem 3D Golf Computerspiel aufnehmen

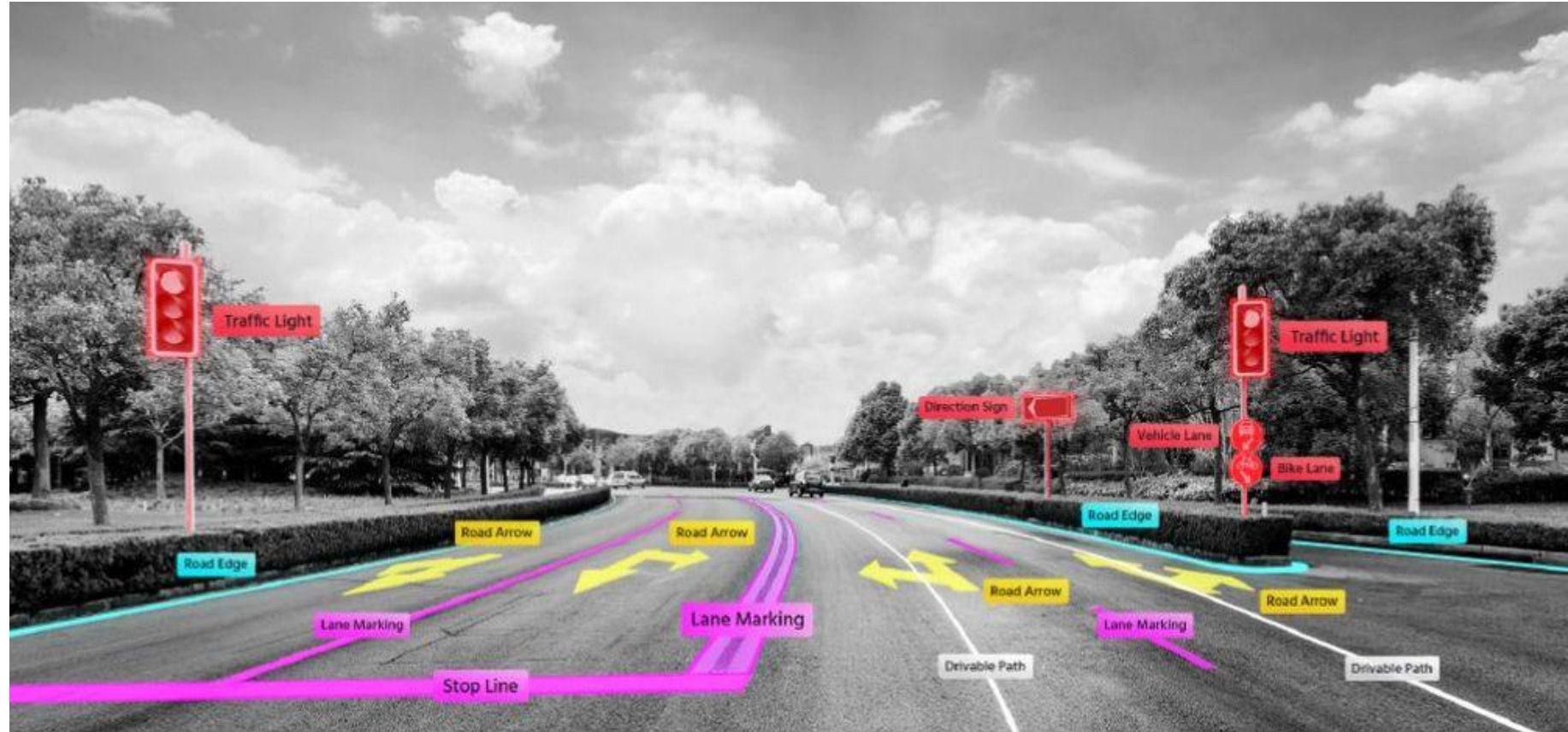
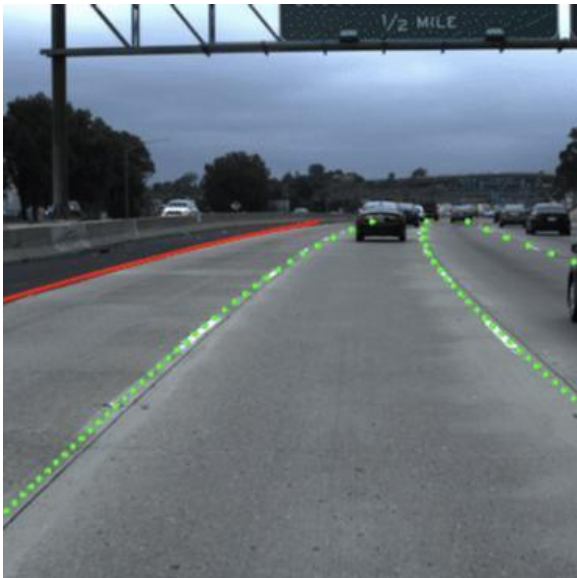
Inhalt

Modellierung der Umgebung von CPS:

- Modellierung mit KI/Deep Learning
 - Objektdetektion, -tracking, Bildsegmentierung, Greifpunkte
 - Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?
 - Spracherkennung
 - Verhaltensmodellierung von Fußgängern, Fahrzeugen
 - World Models
- Lernen in virtuellen Umgebungen (in einer Simulation)
 - 3D Simulation
 - Von der Simulation in die Realität (Sim2real)
- **Statische Modelle der Umgebung (Karte)**
 - **SLAM – Simultaneous Localization and Mapping**
 - Semantische Karte (Karte mit Objekten/Orten: Couch, TV, Küche, ...)

HD Karte für autonomes Fahren

High Definition map



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

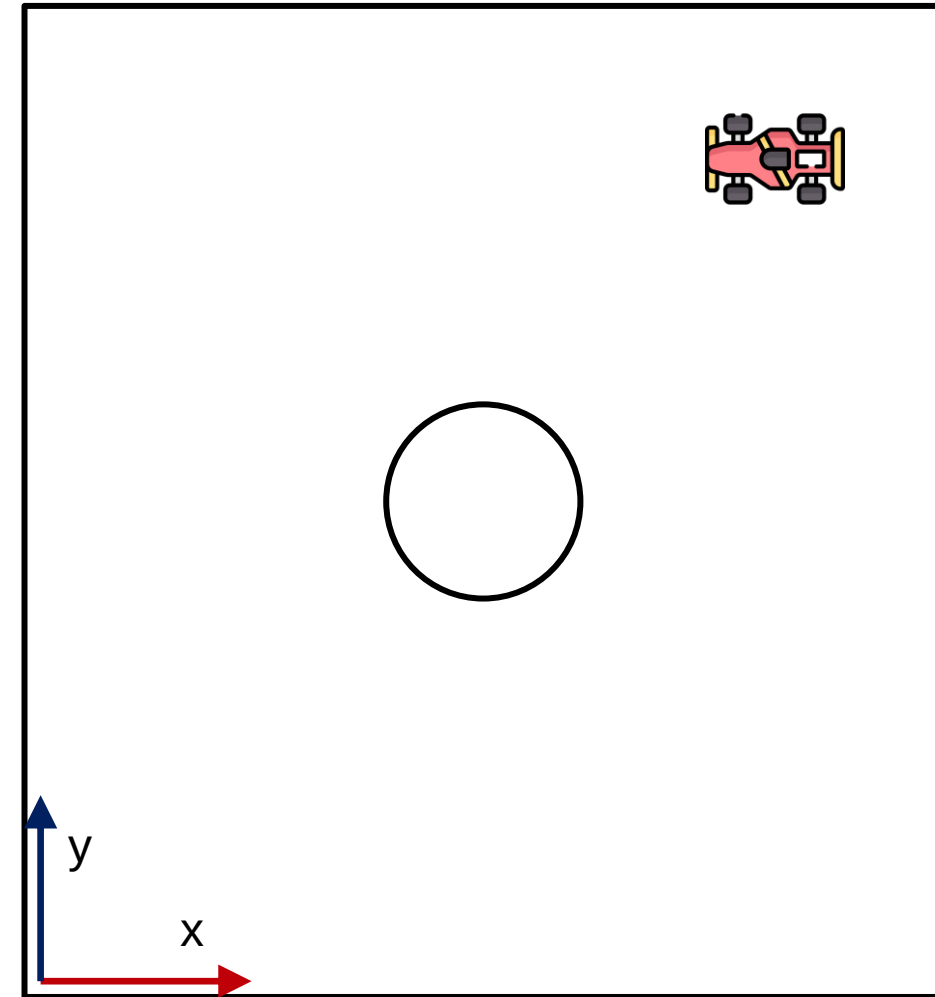
<https://www.mobileye.com/blog/av-maps-vs-hd-maps/>
<https://paperswithcode.com/task/lane-detection>

Technology
Arts Sciences
TH Köln

SLAM - Simultaneous Localization and Mapping

Wenn Sie keine Karte der Umgebung haben...

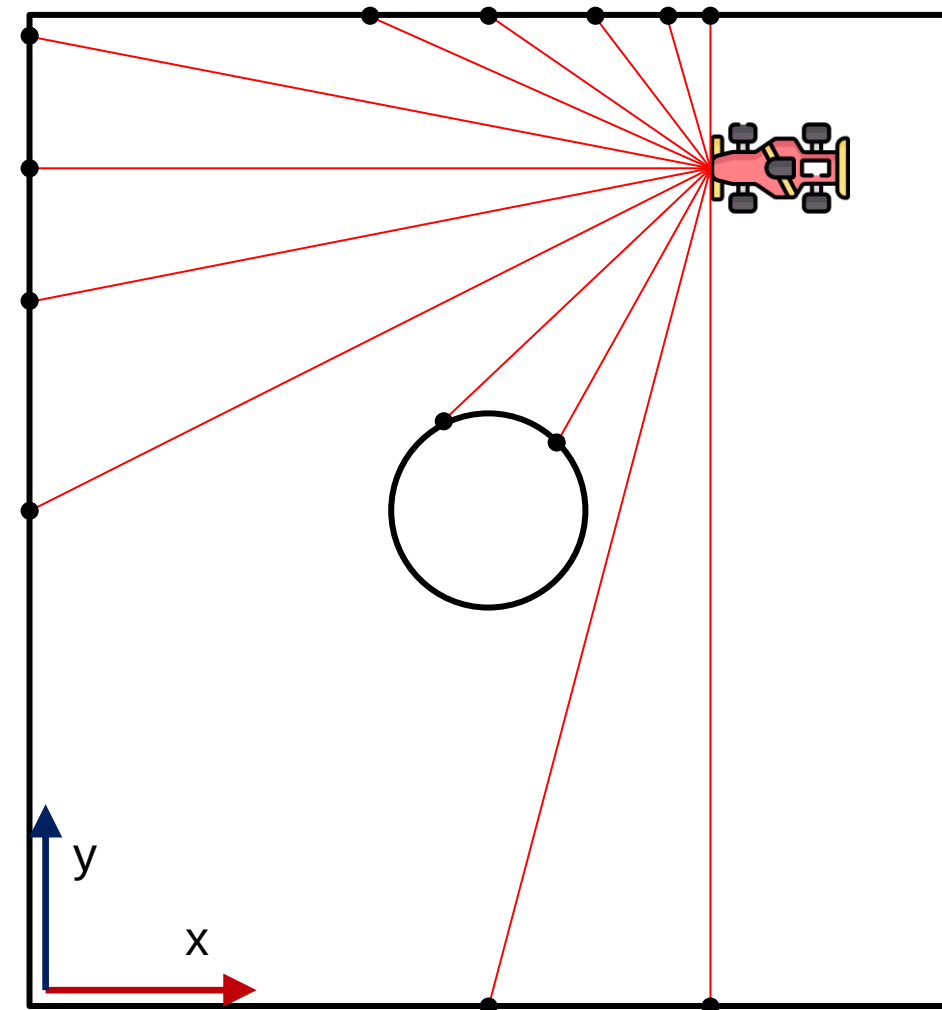
- Erstellung einer Karte der Umgebung durch ein autonom fahrendes System (bspw. Roboter)
- Problem:
 - Roboter weiß selbst nicht wo er sich befindet



SLAM - Simultaneous Localization and Mapping

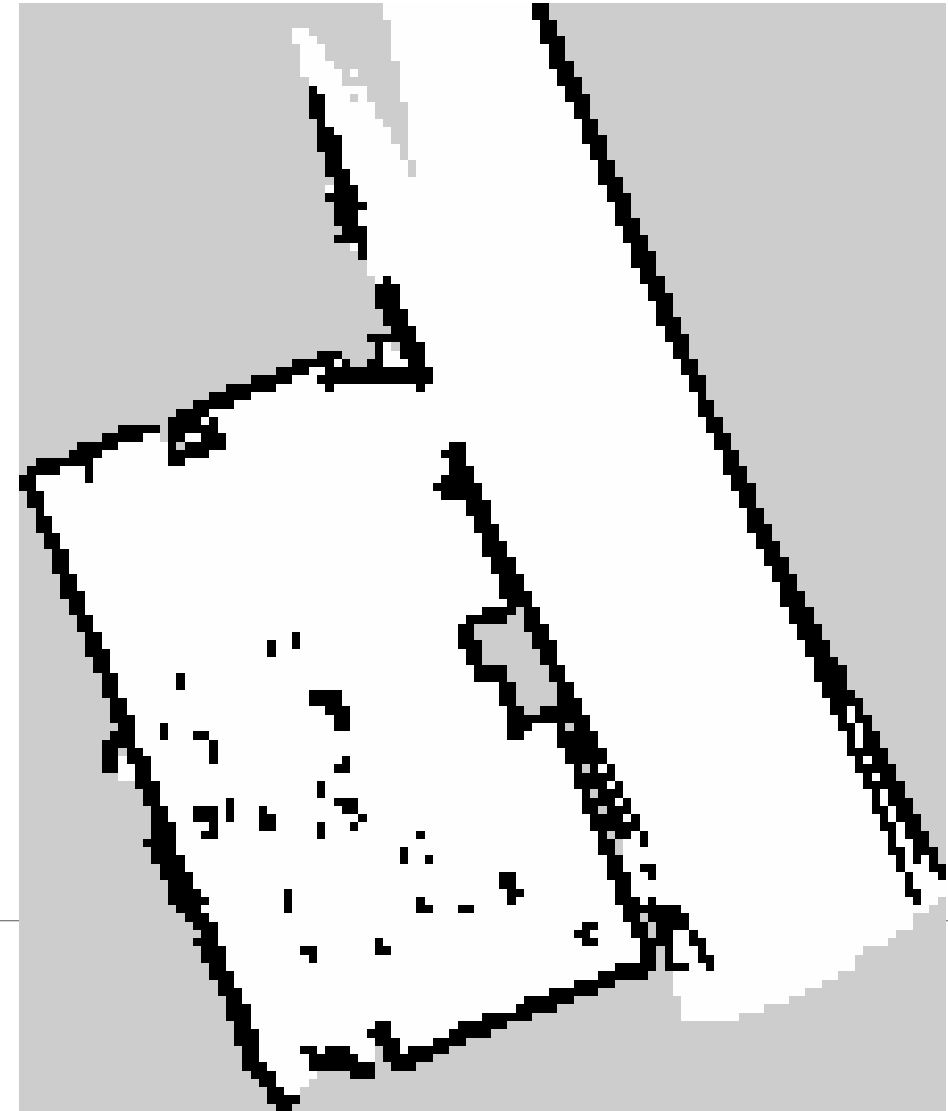
Wenn Sie keine Karte der Umgebung haben...

- Erstellung einer Karte der Umgebung durch ein autonom fahrendes System (bspw. Roboter)
- Problem:
 - Roboter weiß selbst nicht wo er sich befindet



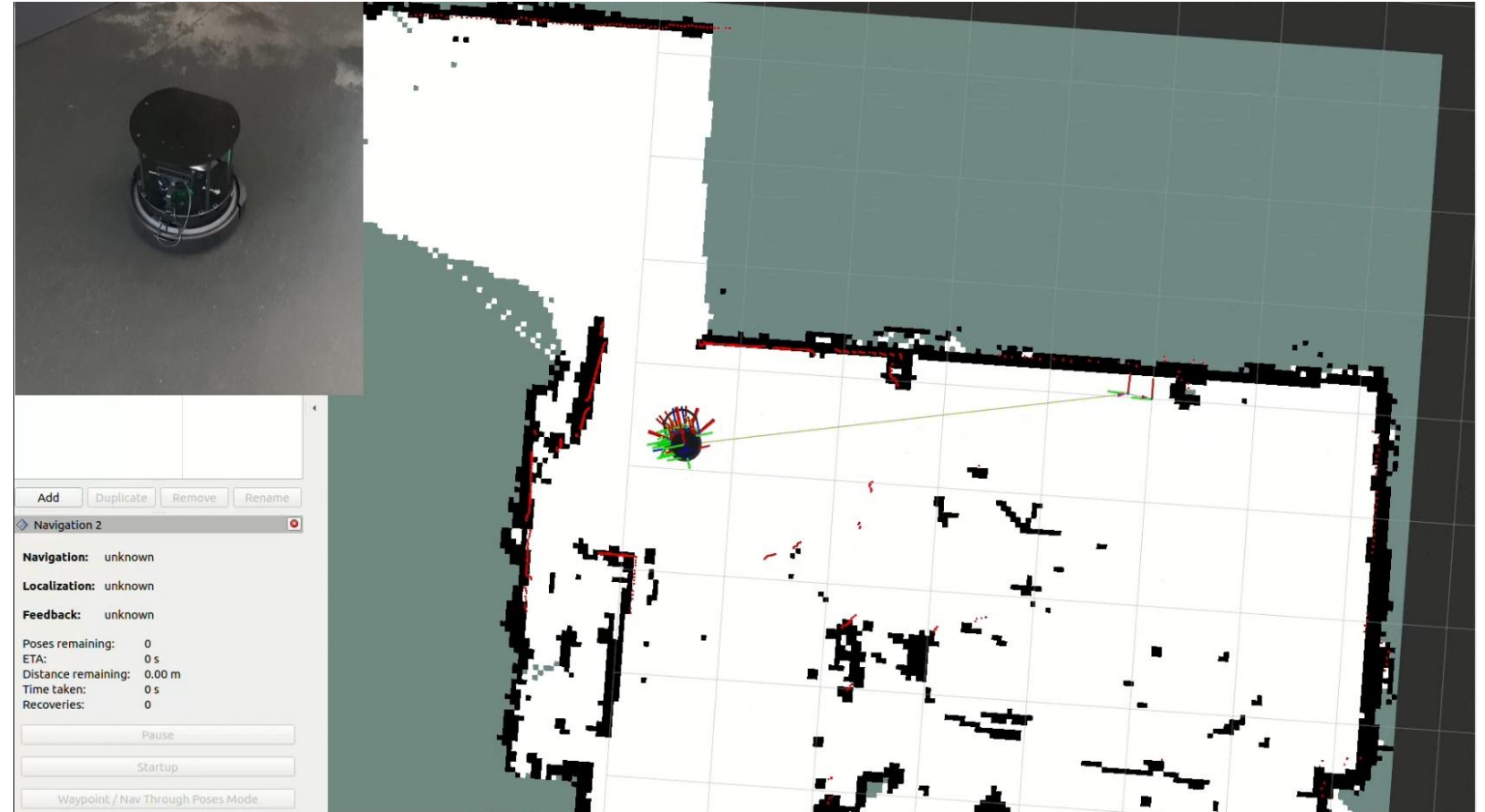
SLAM - Simultaneous Localization and Mapping

Karte des Raumes 1.242 mit Teil des Flures
erstellt durch Turtlebot mit SLAM



SLAM - Simultaneous Localization and Mapping

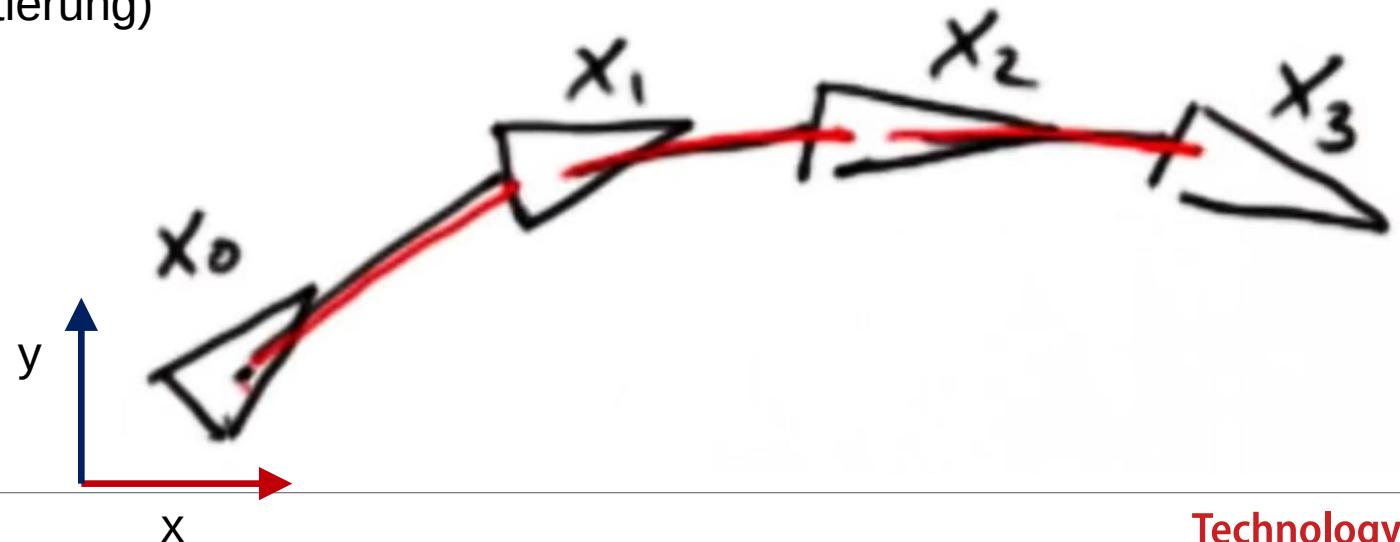
Karte des Flures vor
Raum 1.242 erstellt durch
Turtlebot mit SLAM



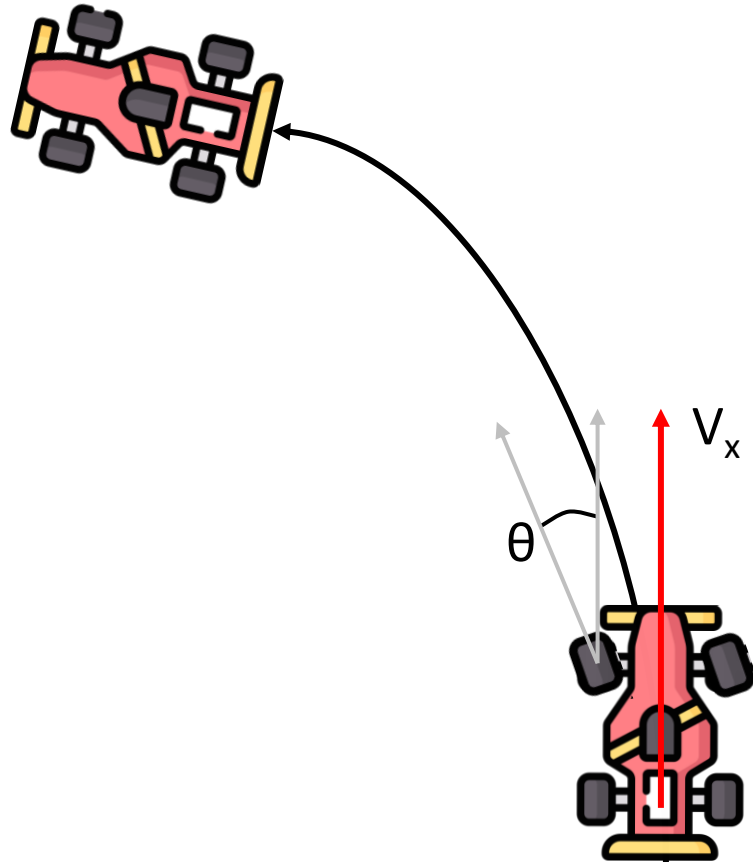
SLAM - Simultaneous Localization and Mapping

Wenn Sie keine Karte der Umgebung haben...

- Erstellung einer Karte der Umgebung durch ein autonom fahrendes System (bspw. Roboter)
- Problem:
 - Roboter weiß selbst nicht wo er sich befindet
- Roboter hat Sensoren mit denen er seine Bewegung messen kann. Bspw.:
 - Odometer (misst lineare Geschwindigkeit)
 - Inertialmesssystem (misst Orientierung)



Dead Reckoning (Koppelnavigation)

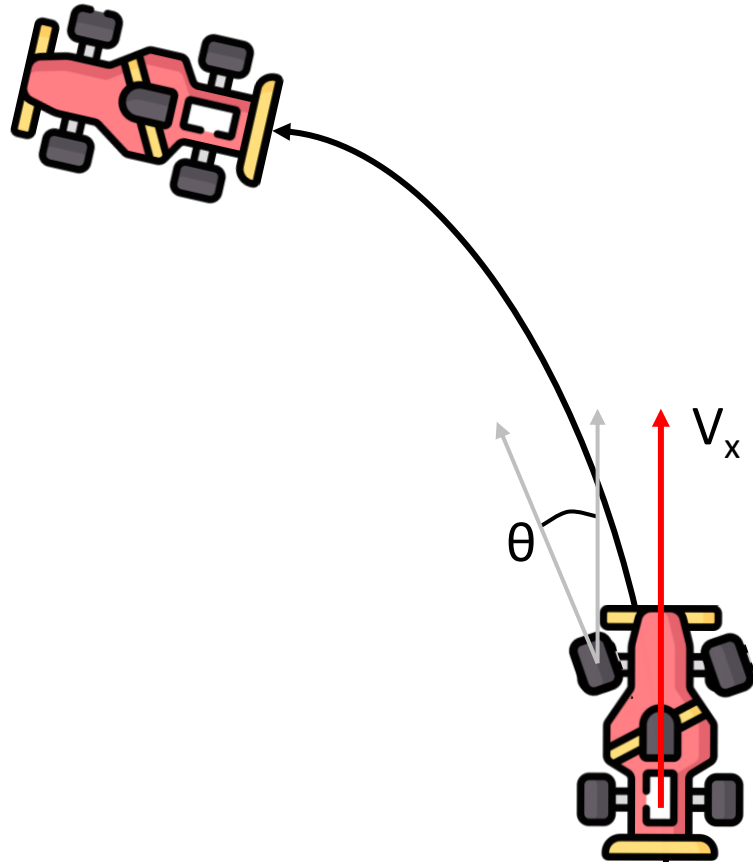


Berechnung der aktuellen Position eines fahrenden Roboters auf der Grundlage einer zuvor ermittelten Position.

Mit Hilfe eines grundlegenden Kinematikmodells kann die relative Position des Fahrzeugs anhand von Messungen

- der Fahrgeschwindigkeit in Längsrichtung und
 - des aktuellen Lenkwinkels
- bestimmt werden.

Dead Reckoning (Koppelnavigation)



Berechnung der aktuellen Position eines fahrenden Roboters auf der Grundlage einer zuvor ermittelten Position.

Mit Hilfe eines grundlegenden Kinematikmodells kann die relative Position des Fahrzeugs anhand von Messungen

- der Fahrgeschwindigkeit in Längsrichtung und
 - des aktuellen Lenkwinkels
- bestimmt werden.

Problem: Bestimmte Position wird mit der Zeit immer ungenauer.

SLAM - Simultaneous Localization and Mapping

Graph SLAM

- Landmark – Fixpunkt, dessen Position bekannt ist

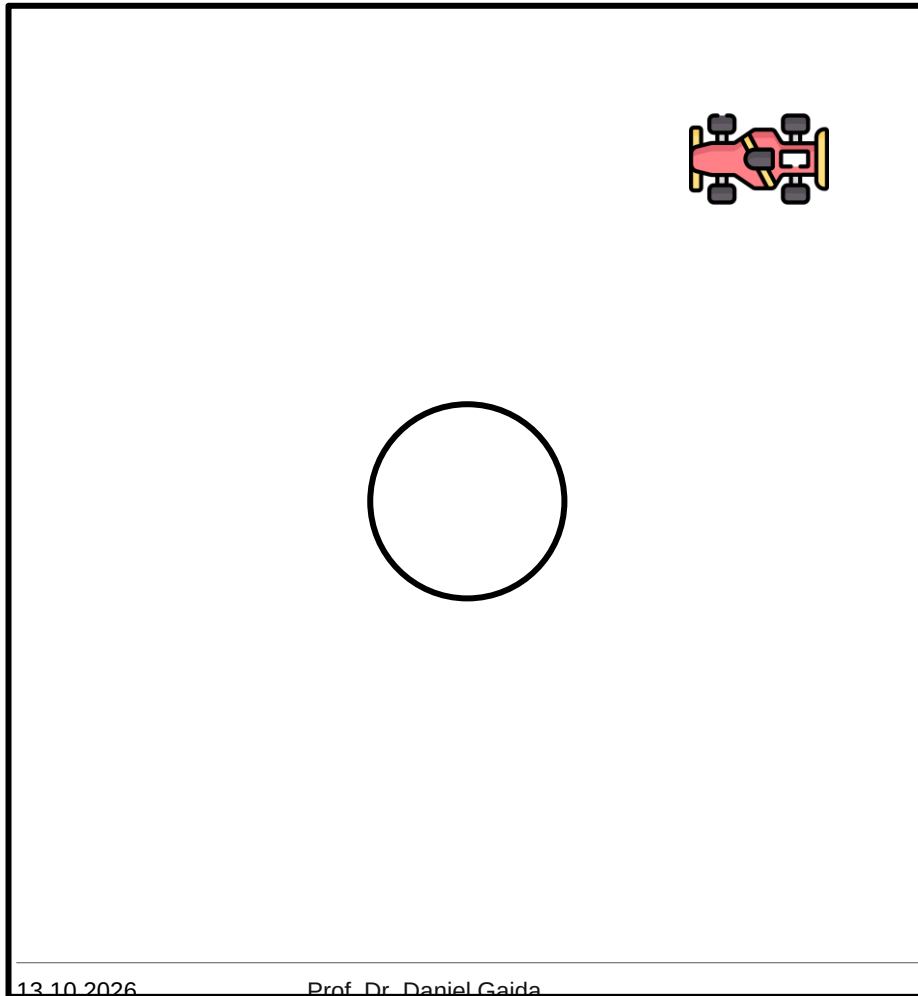
SLAM - Simultaneous Localization and Mapping

Graph SLAM

- Landmark – Fixpunkt, dessen Position bekannt ist



Sensing the environment



13.10.2026

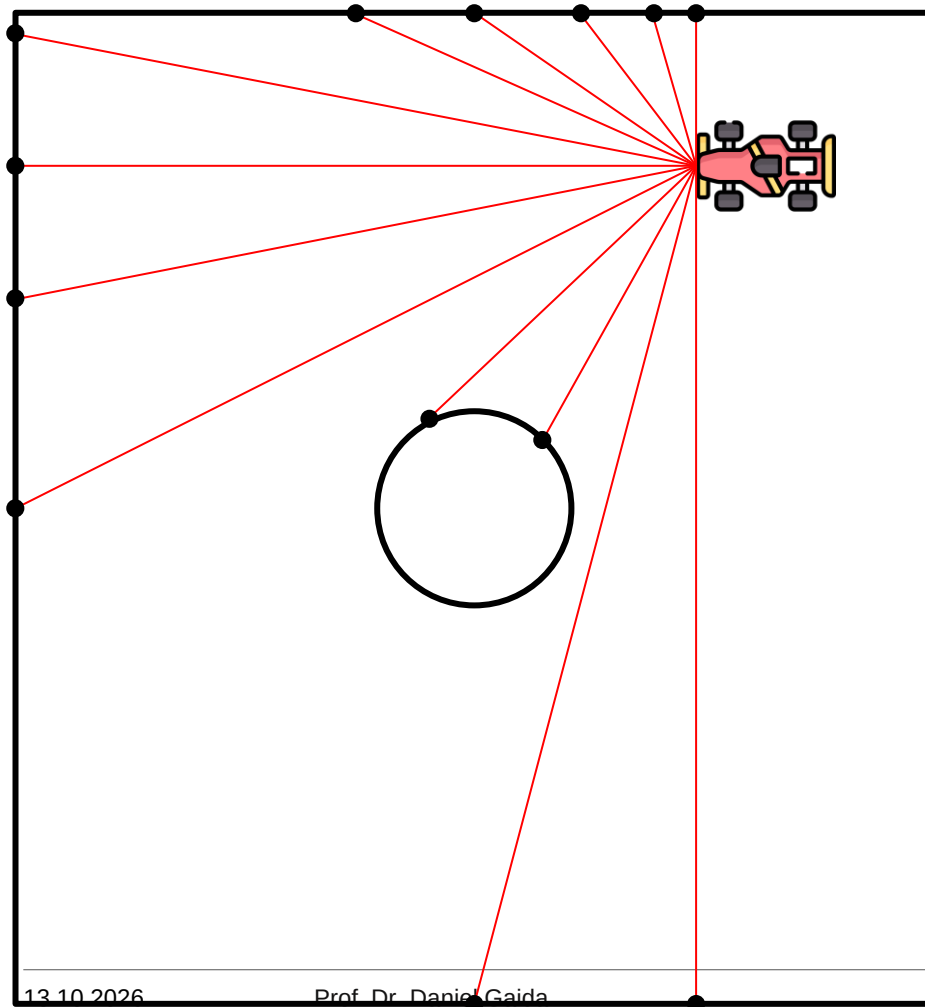
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Sensing the environment



13.10.2026

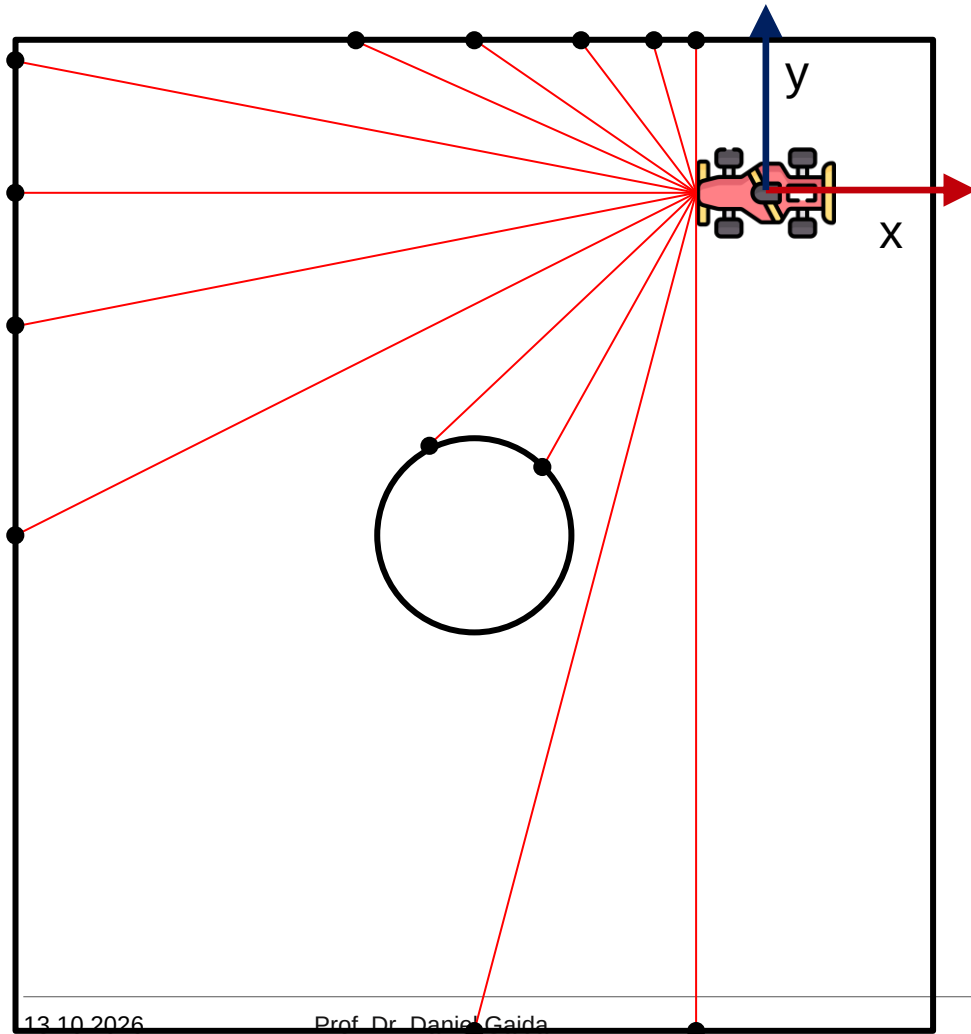
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Sensing the environment



13.10.2026

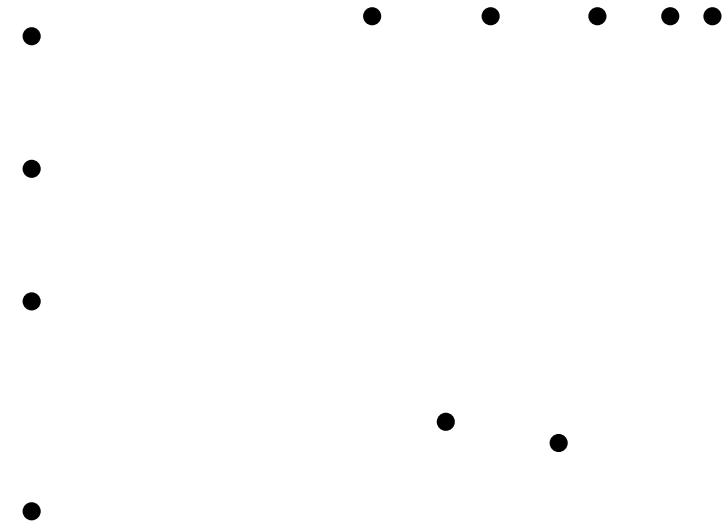
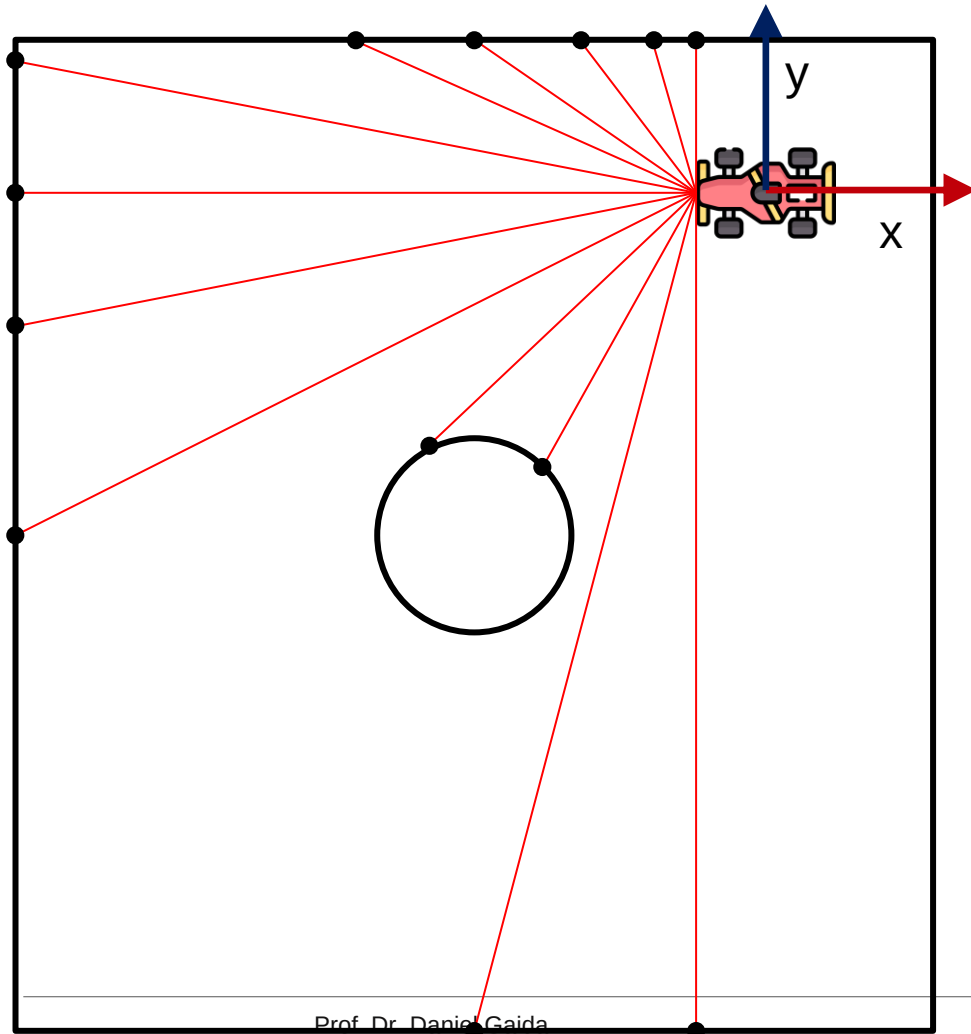
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Sensing the environment



Prof. Dr. Daniel Gaida

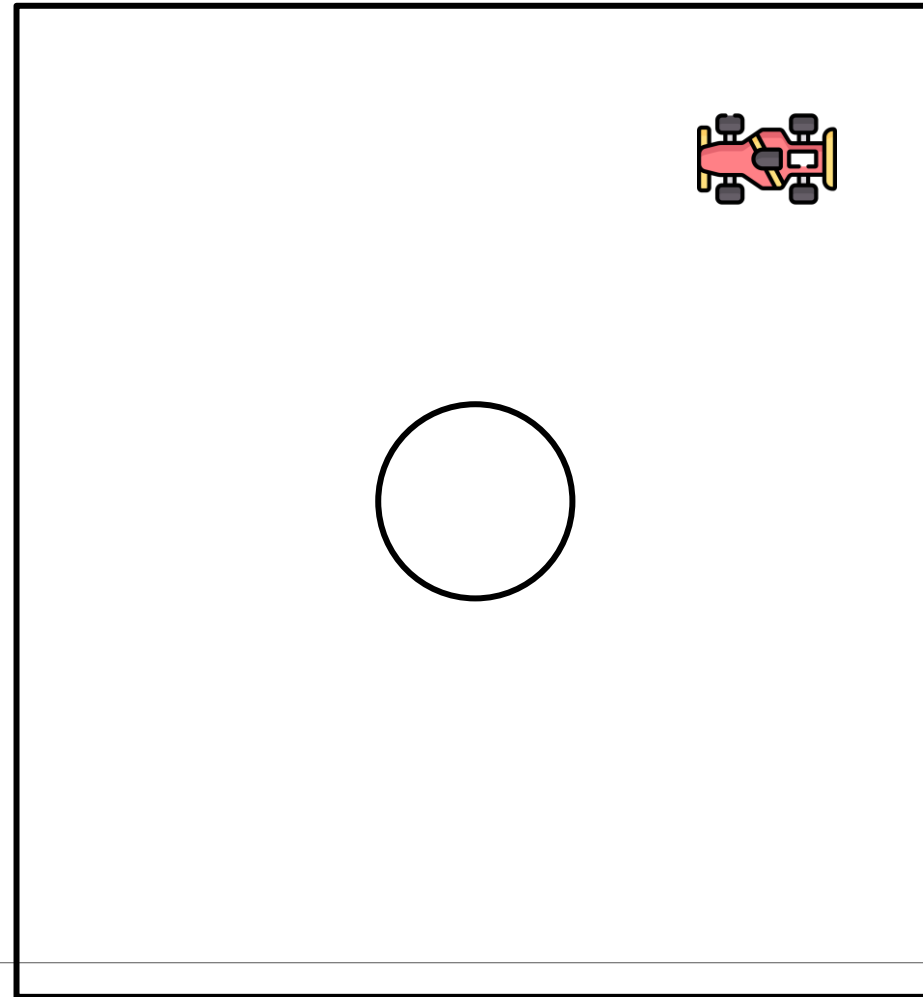
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

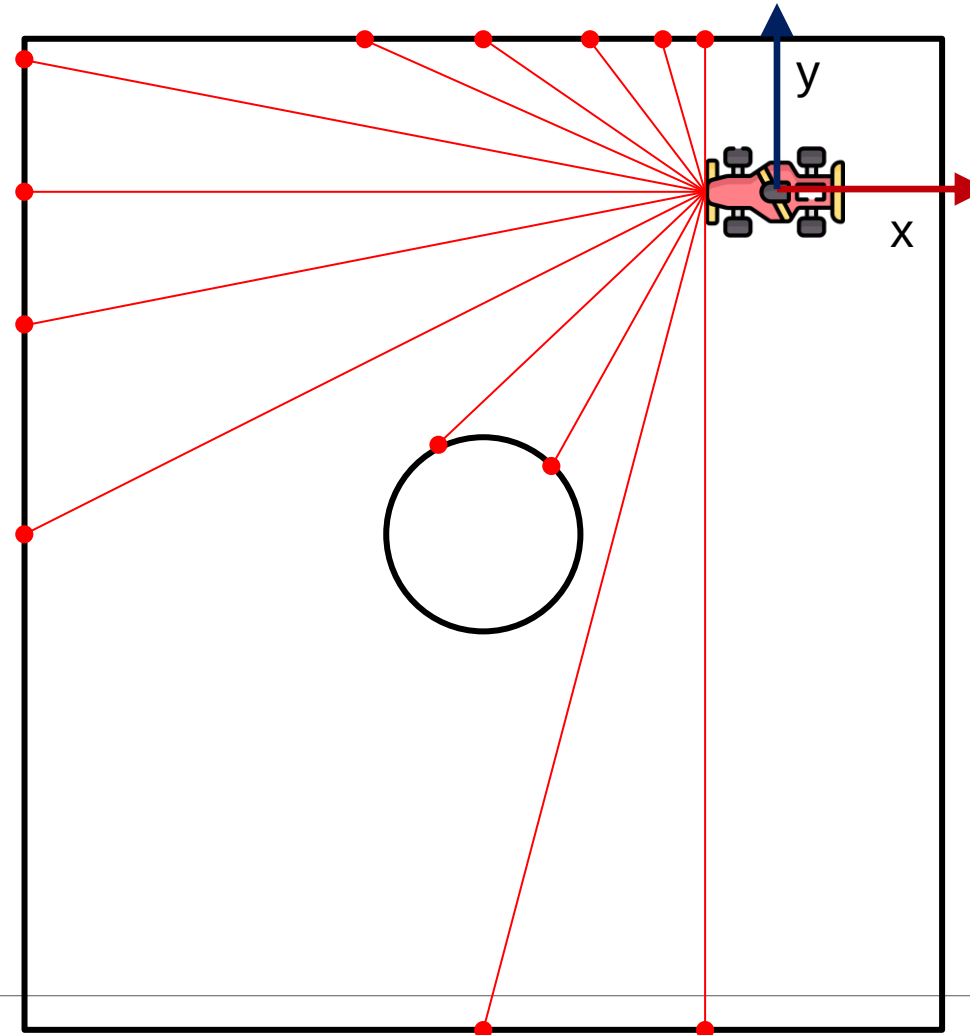
R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Alright, let's try to make a map.

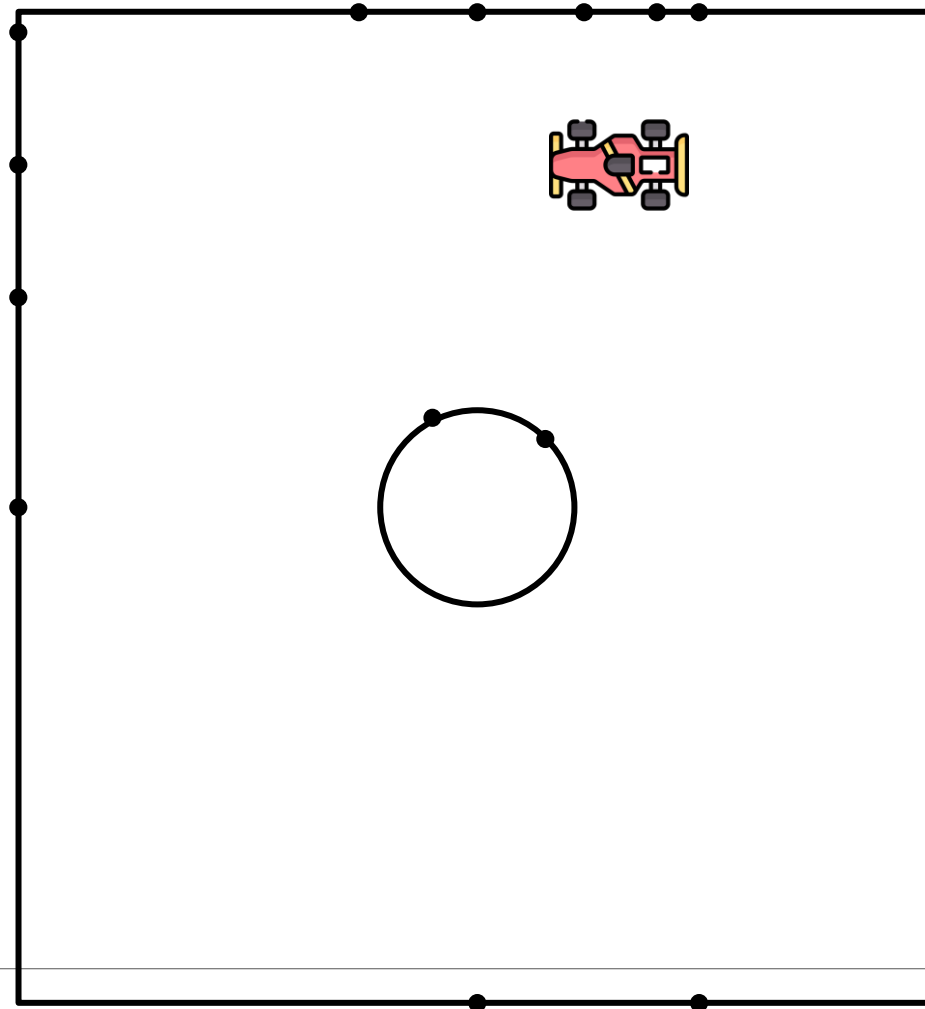
In an ideal world...



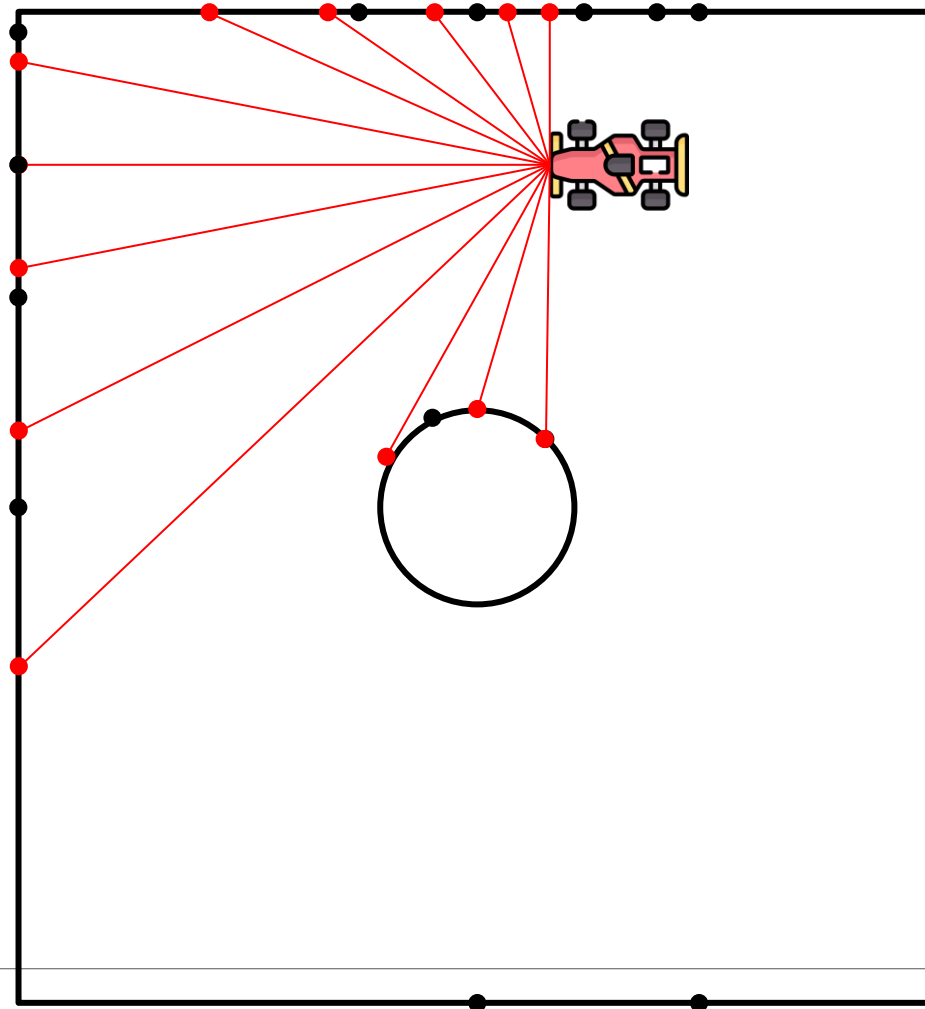
In an ideal world...



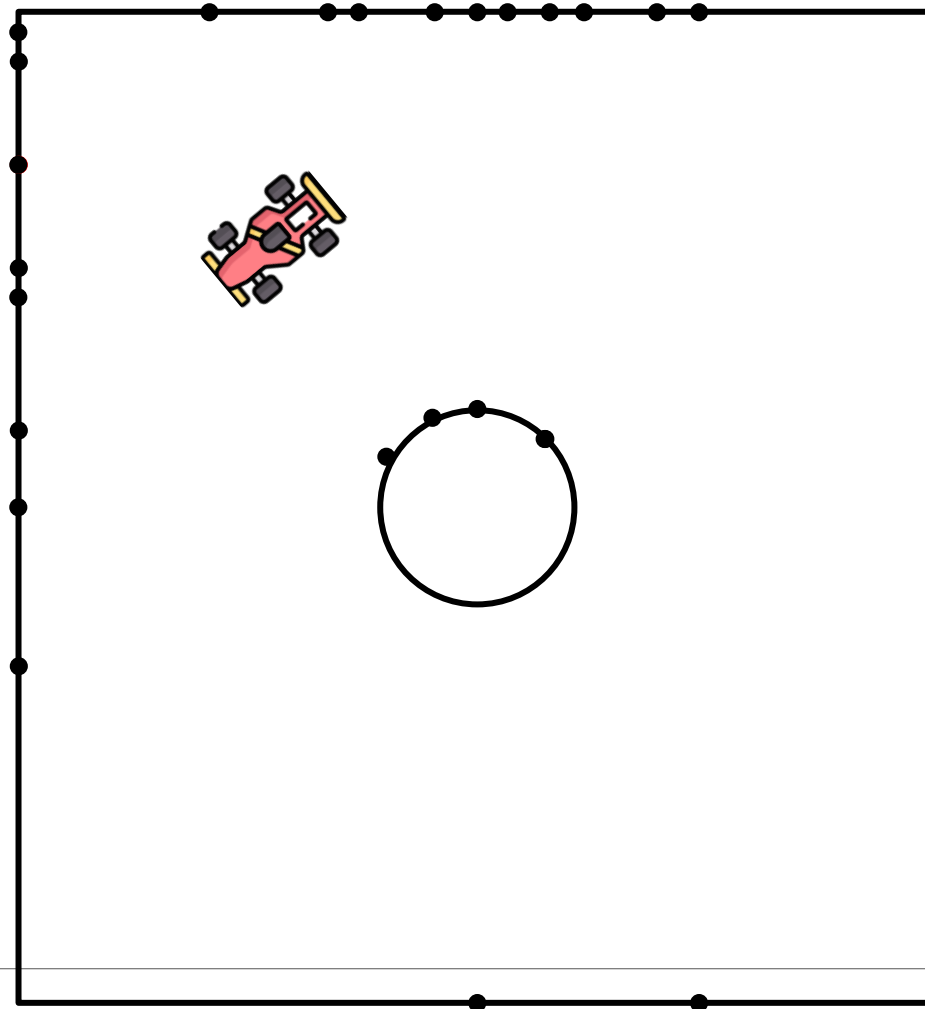
In an ideal world...



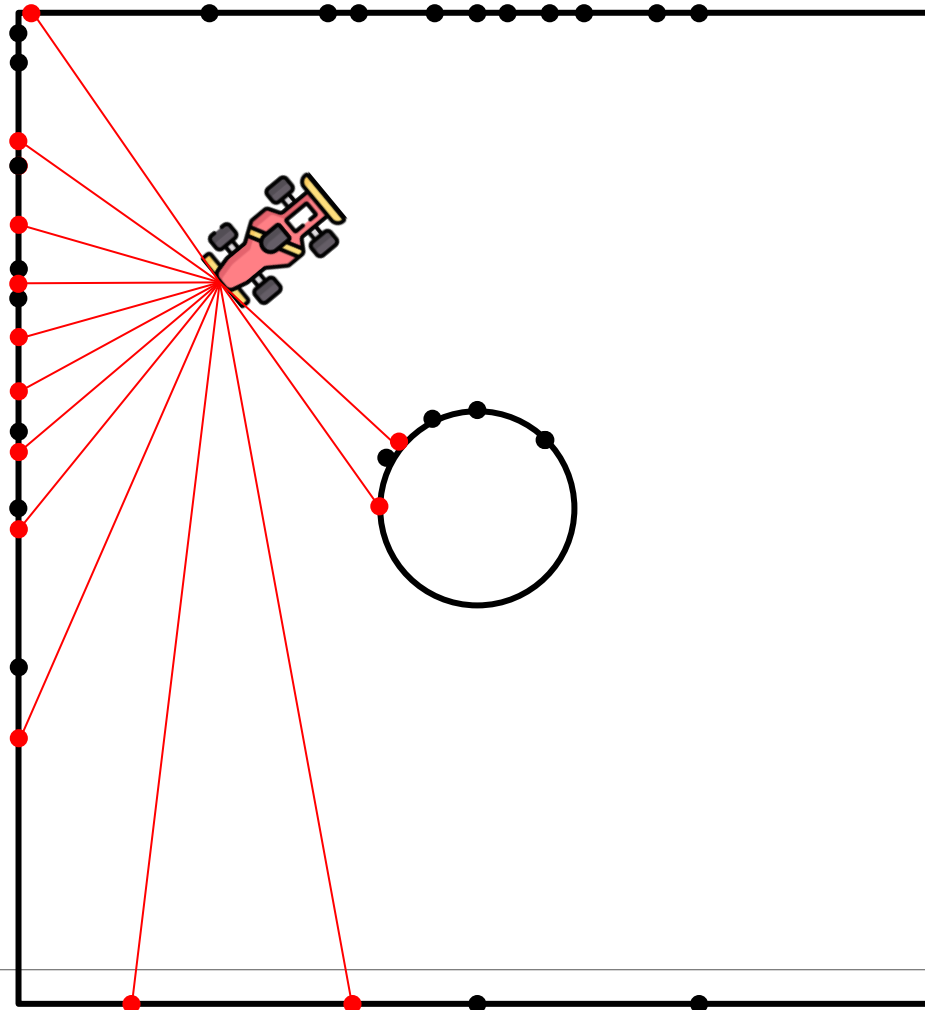
In an ideal world...



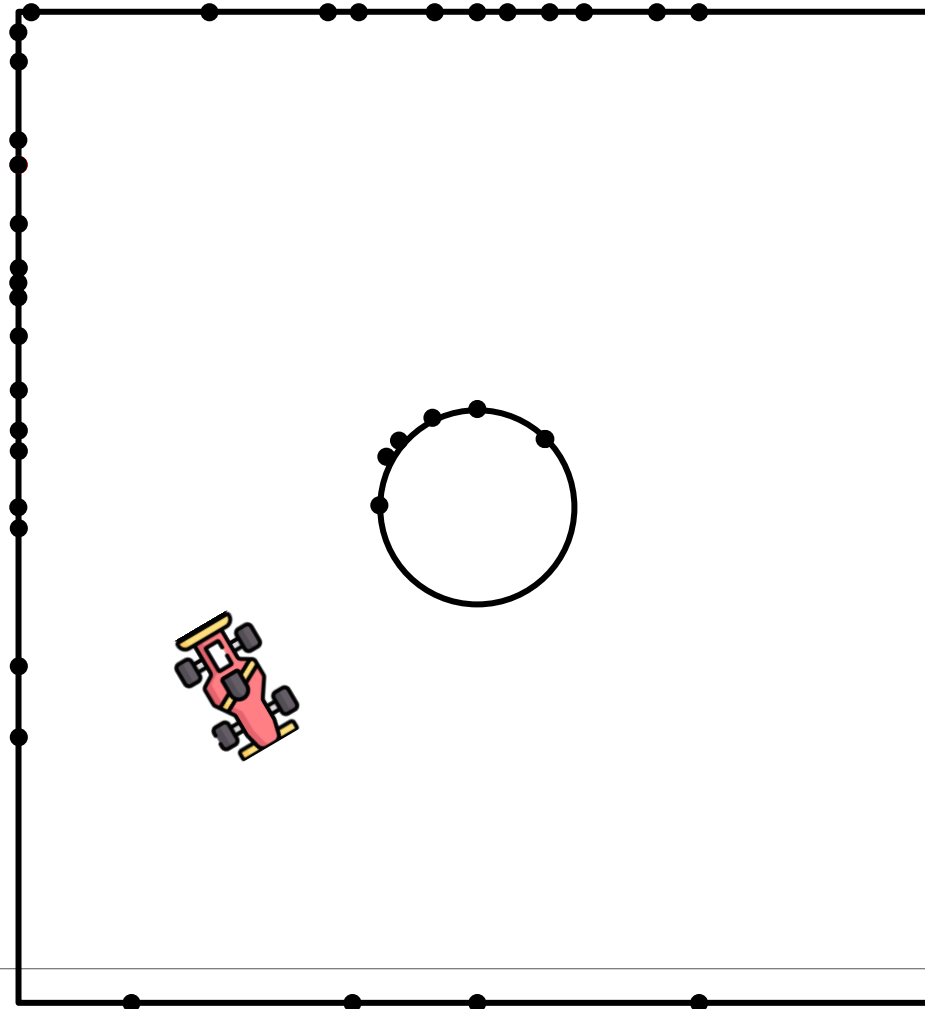
In an ideal world...



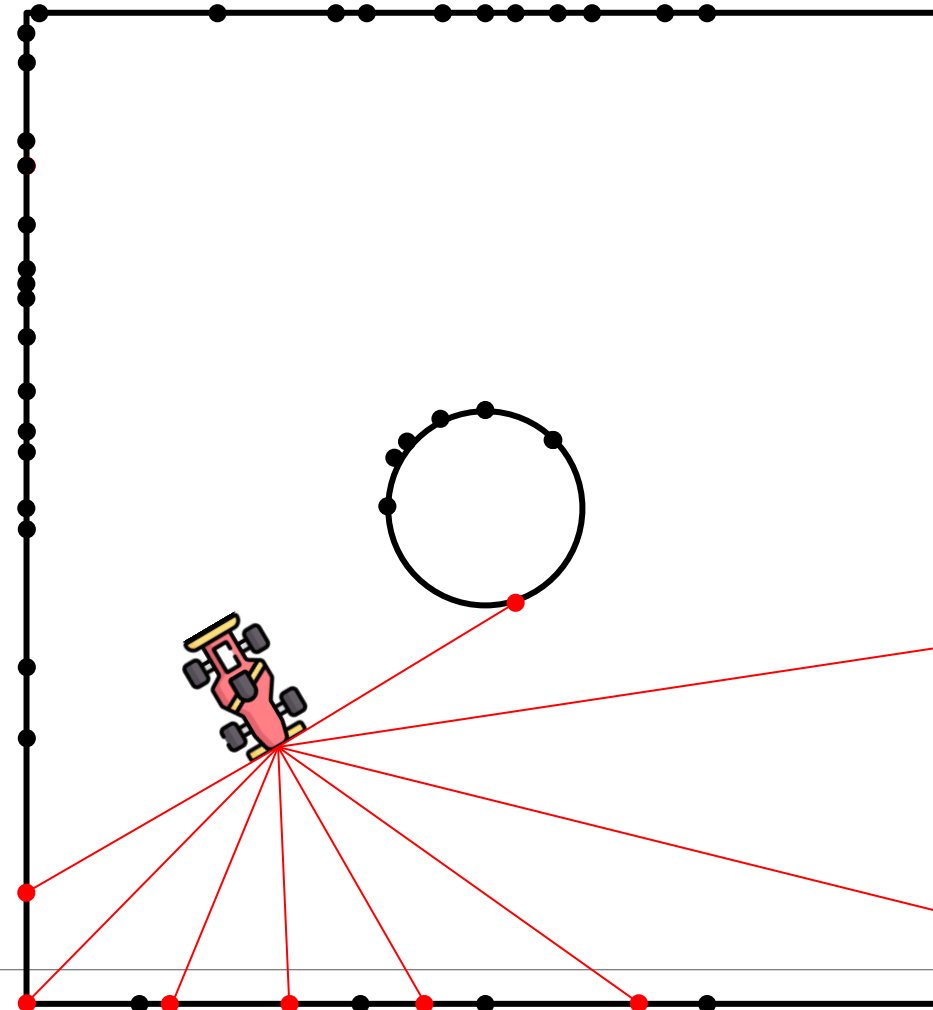
In an ideal world...



In an ideal world...



In an ideal world...



13.10.2026

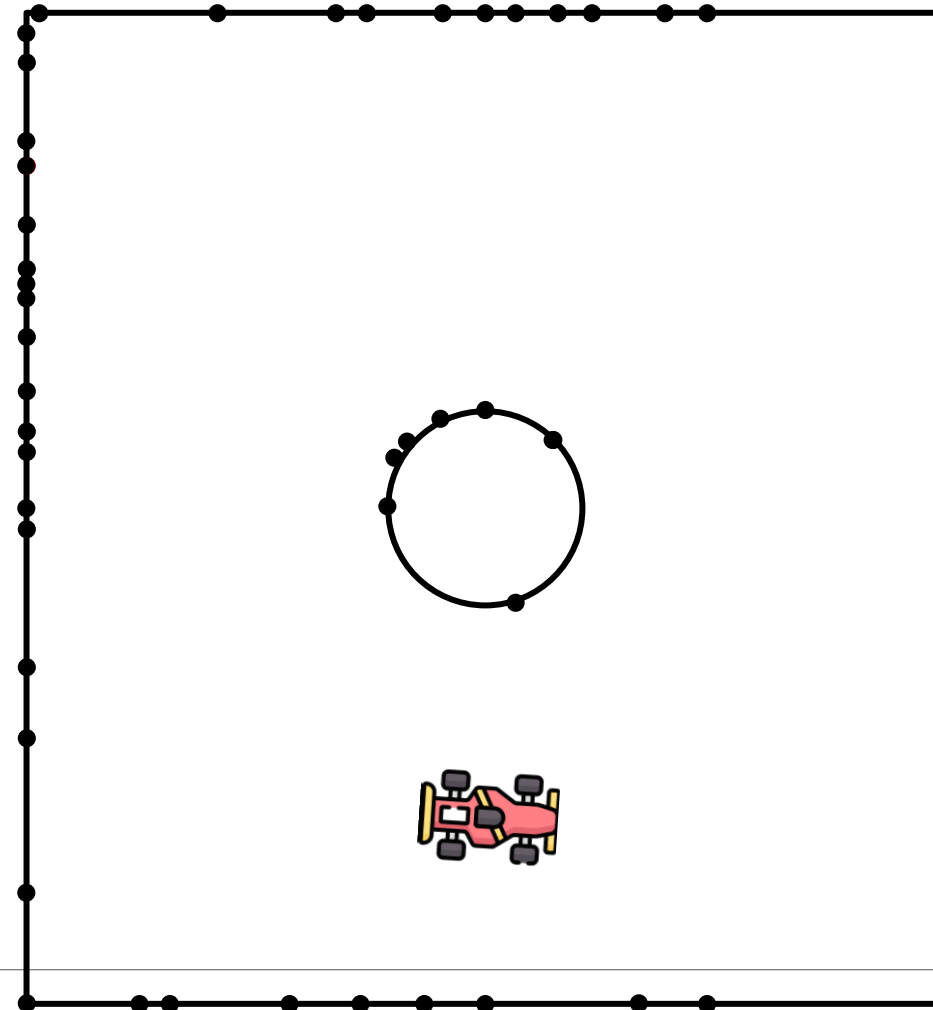
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

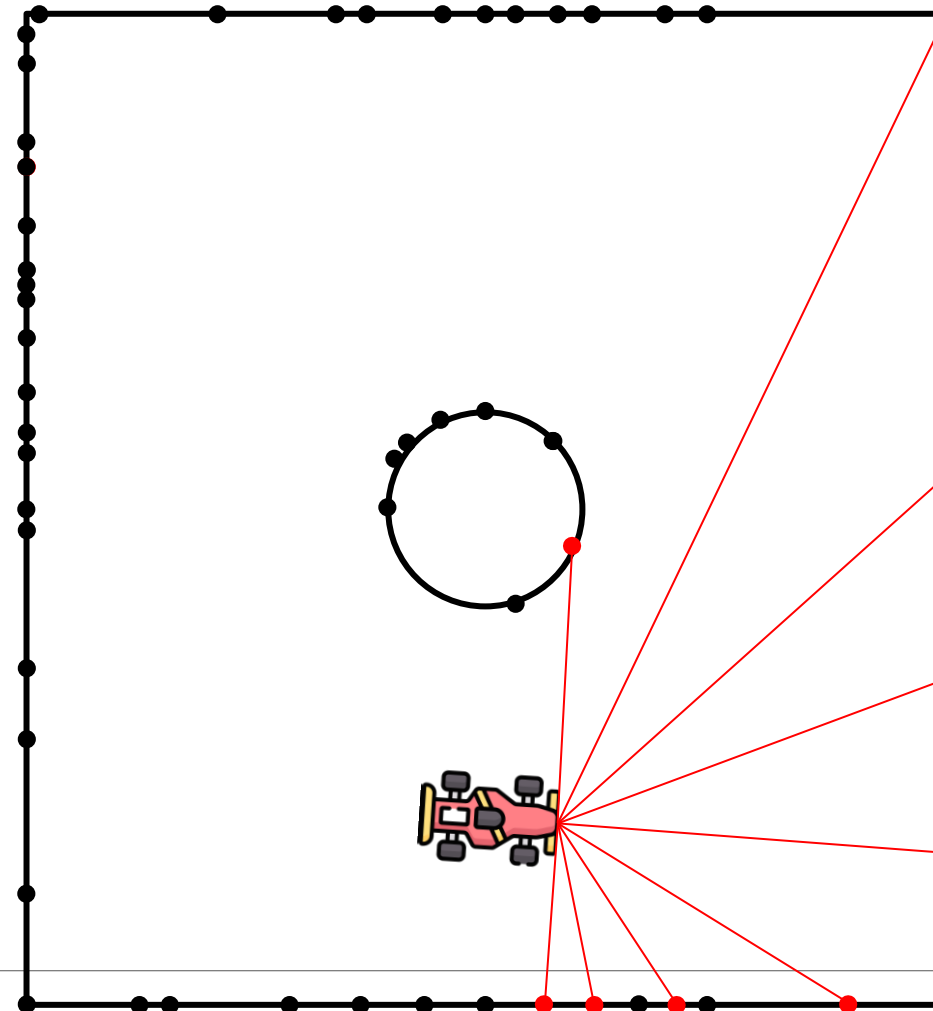
Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

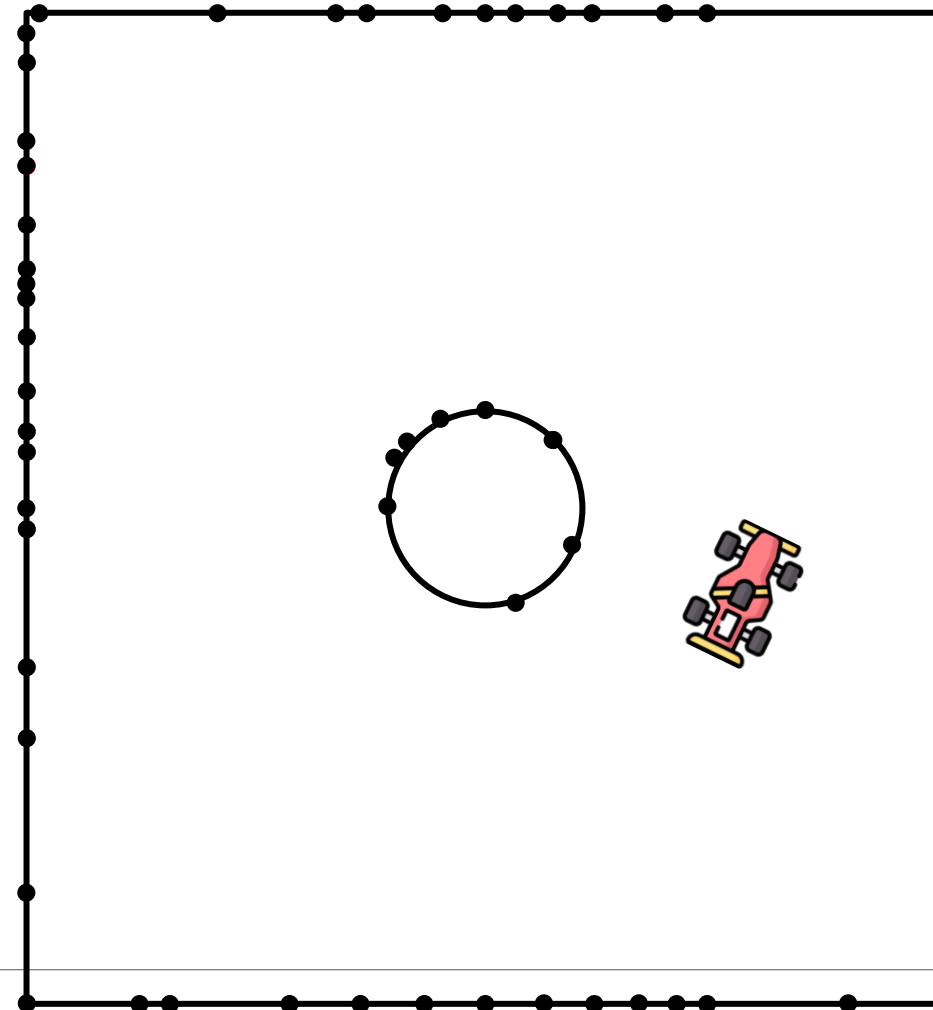
In an ideal world...



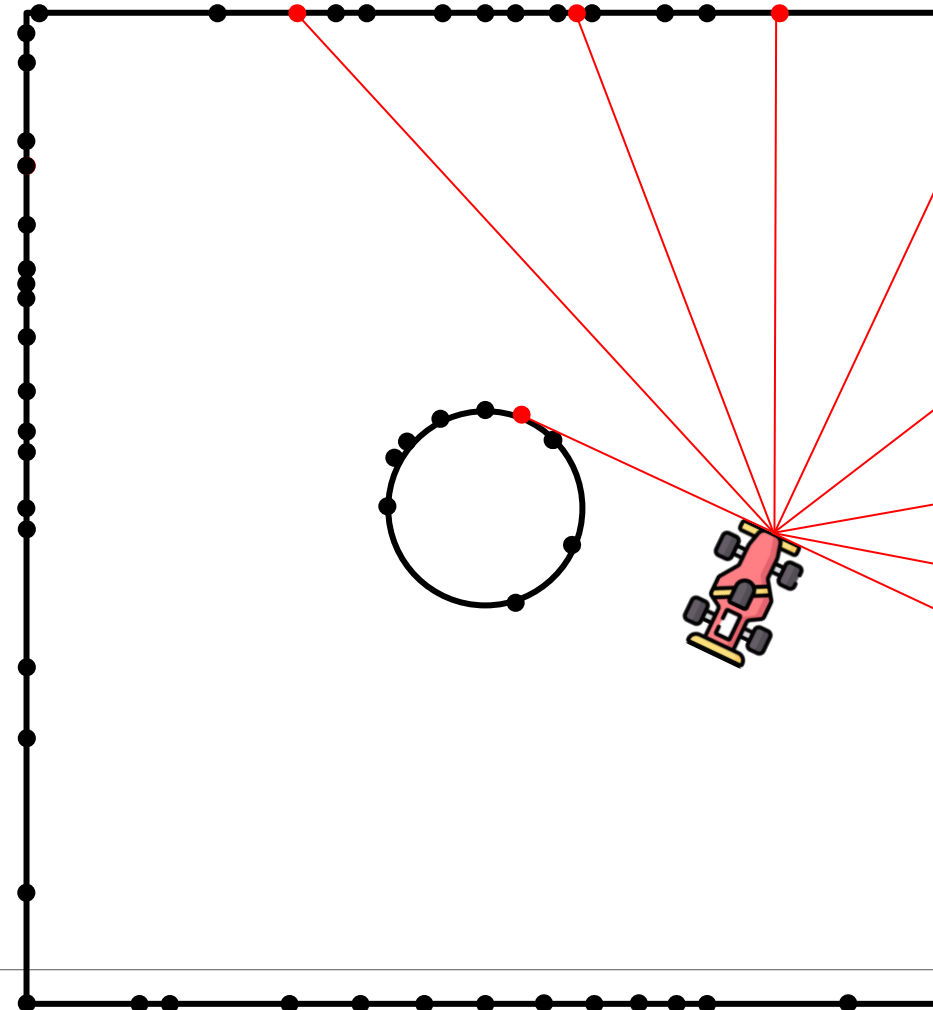
In an ideal world...



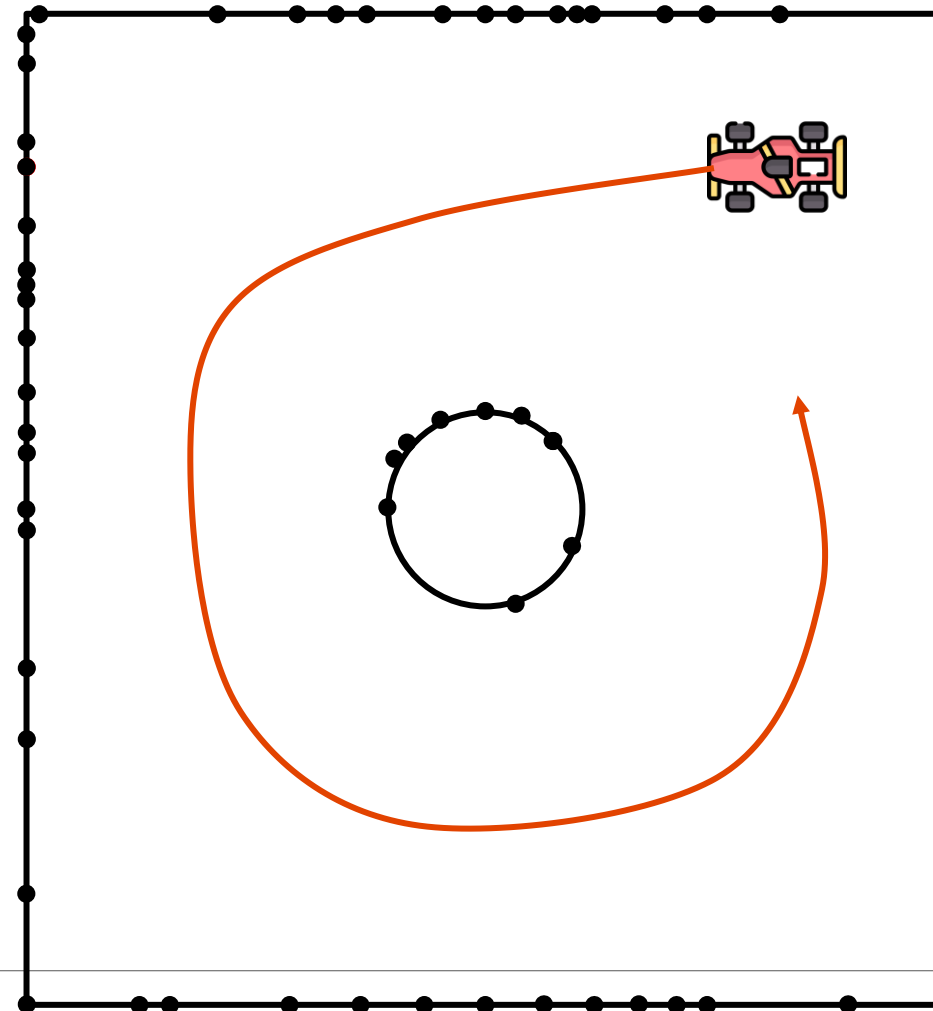
In an ideal world...



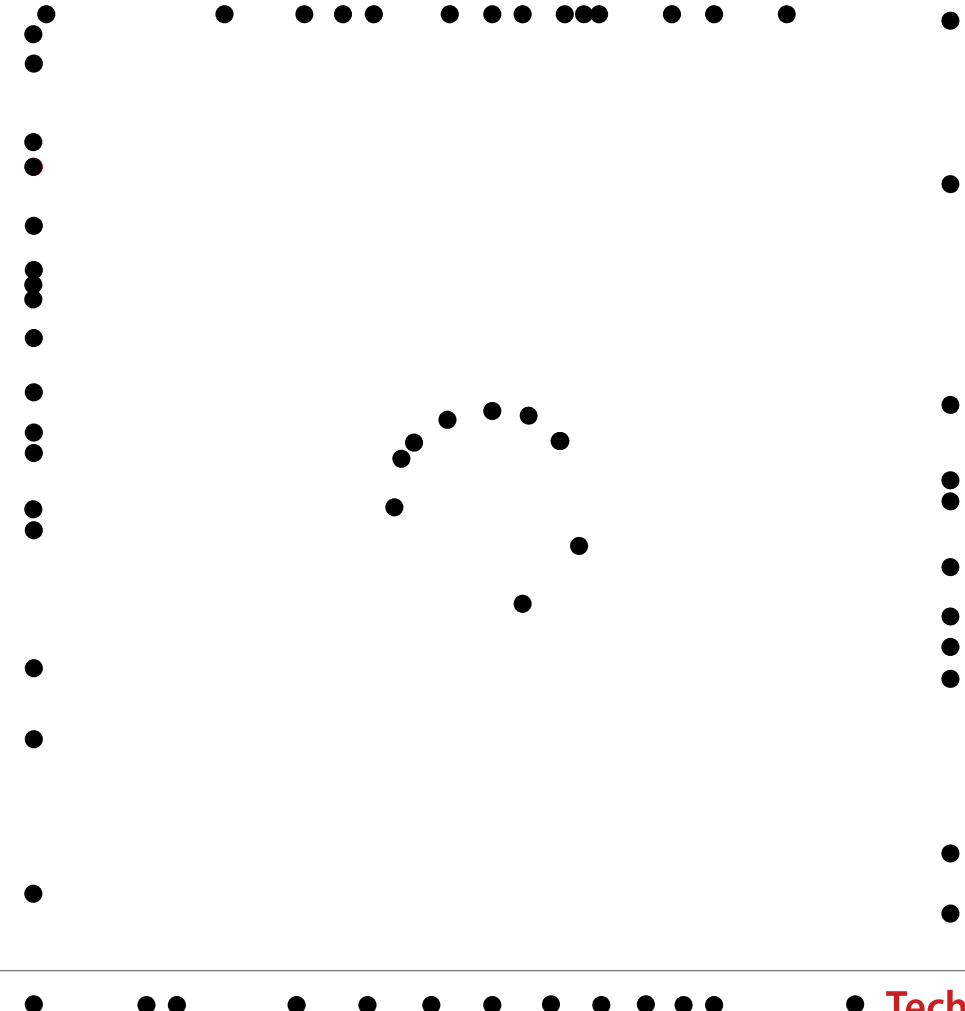
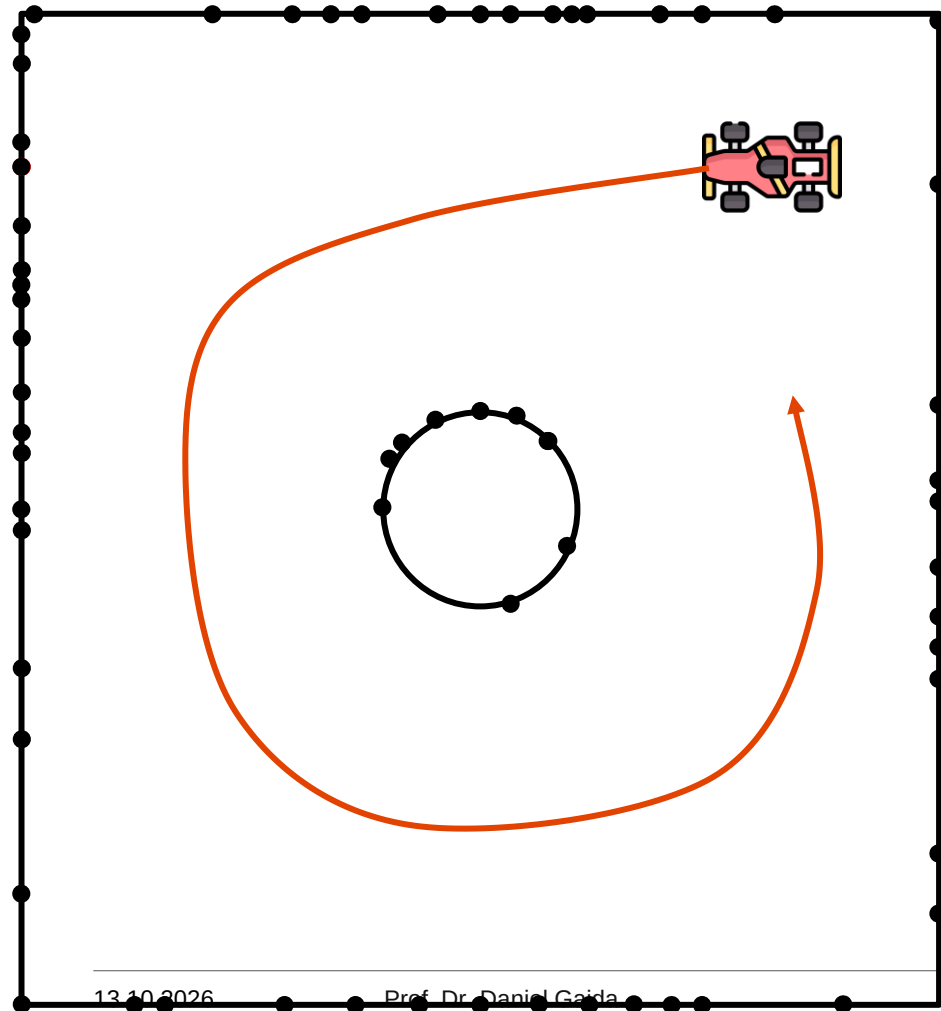
In an ideal world...



In an ideal world...



In an ideal world...



13.10.2026

Prof. Dr. Daniel Gaida

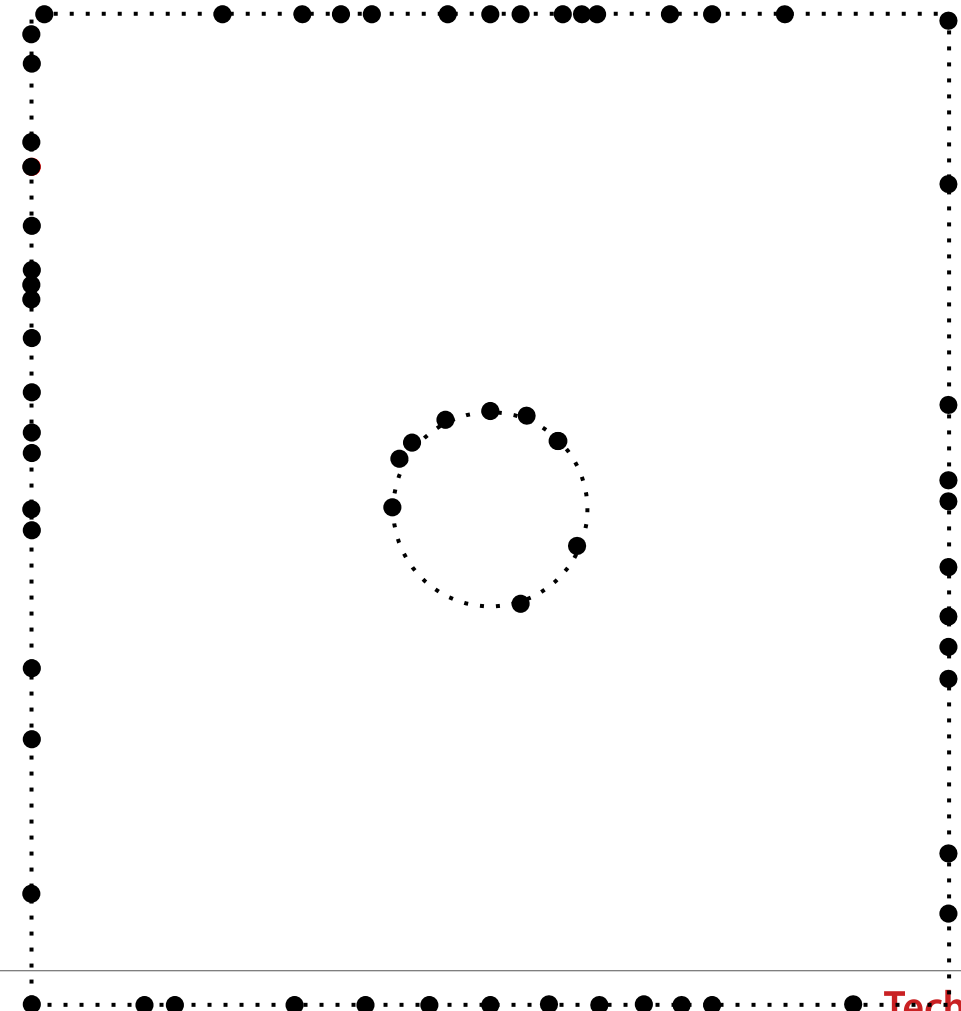
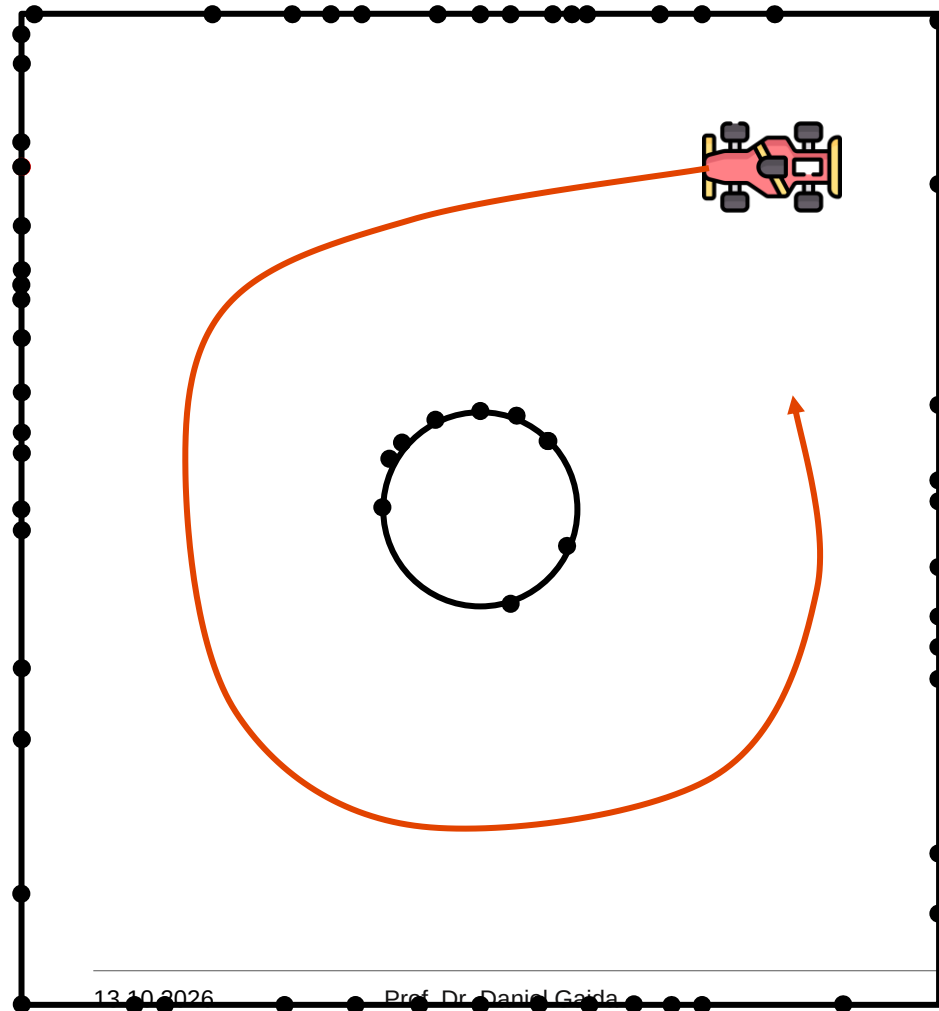
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

• Technology
Arts Sciences
TH Köln

In an ideal world...



13.10.2026

Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

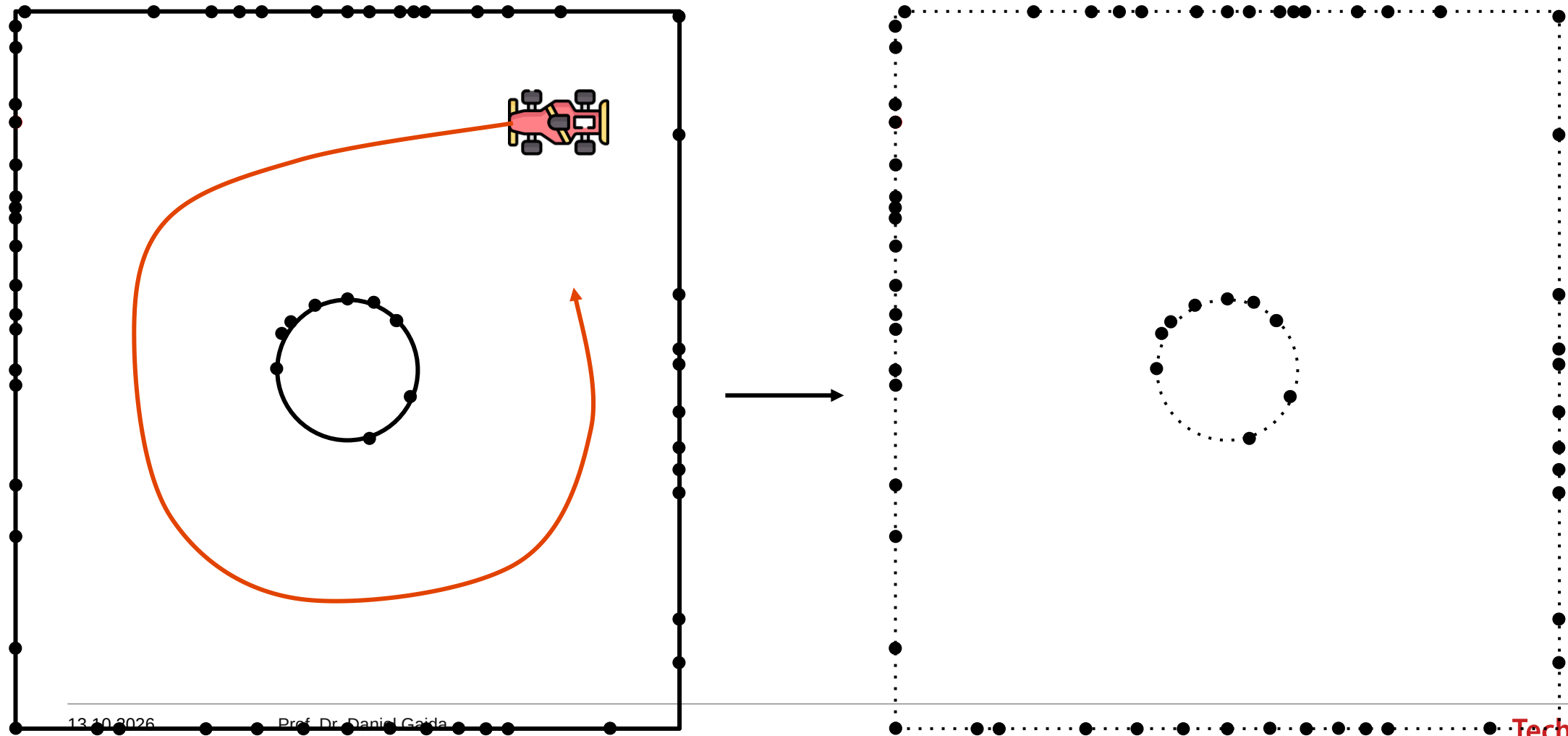
Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

Wie lokalisieren wir uns in dieser Umgebung?

Partikelfilter: Vergleich Lidar-Messung mit Karte



13.10.2026

Prof. Dr. Daniel Gaida

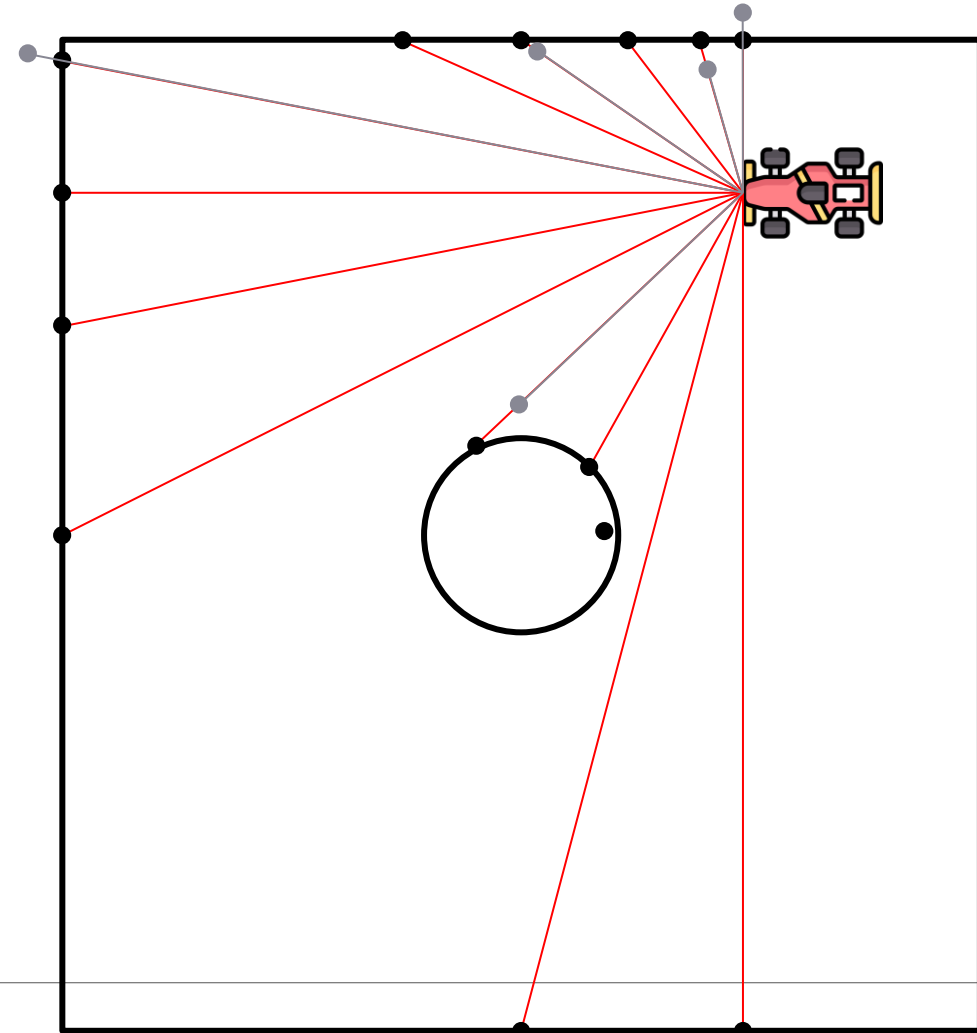
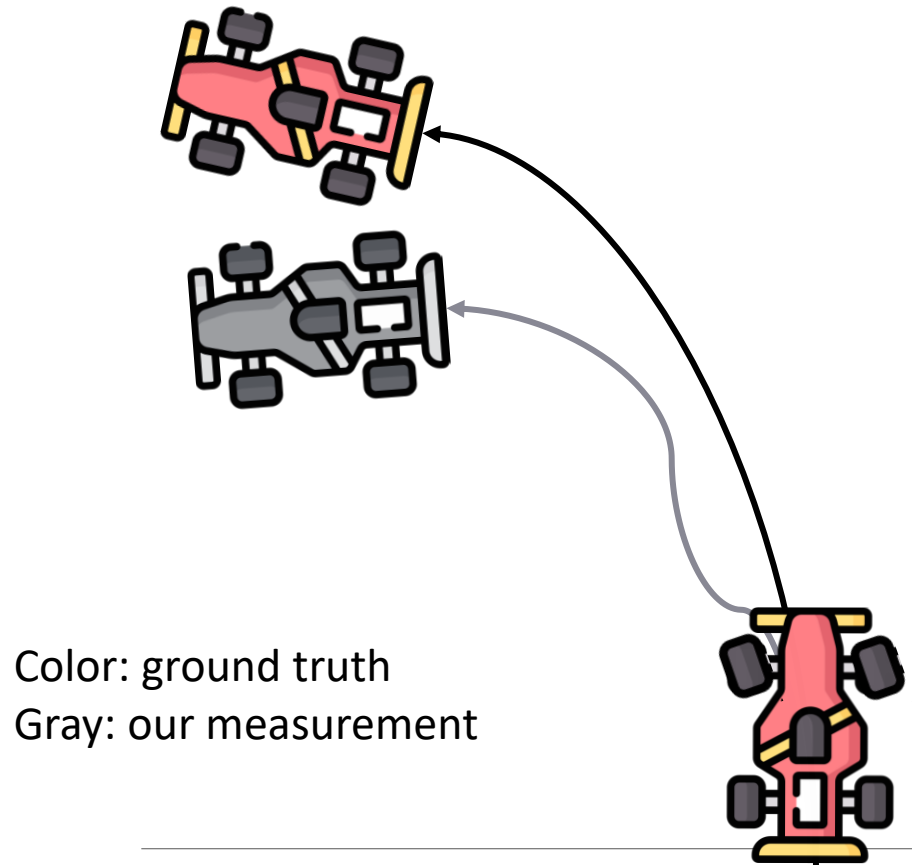
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

But in the real world...



13.10.2026

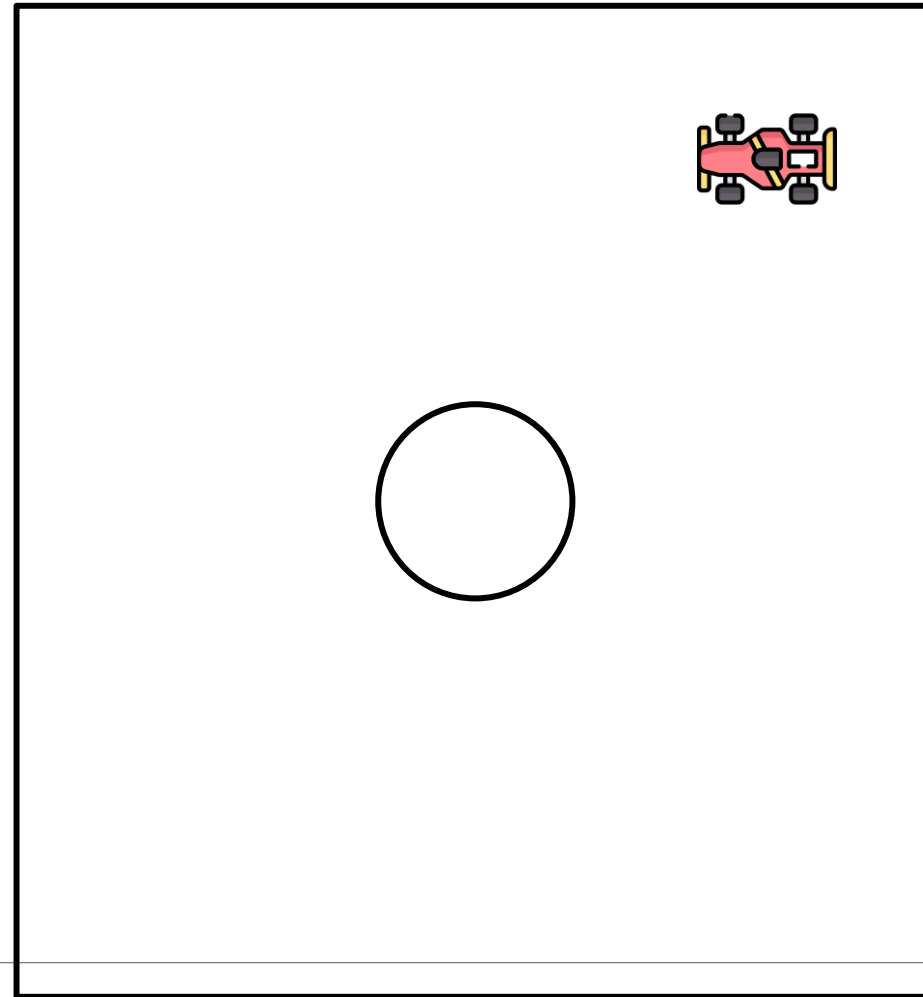
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

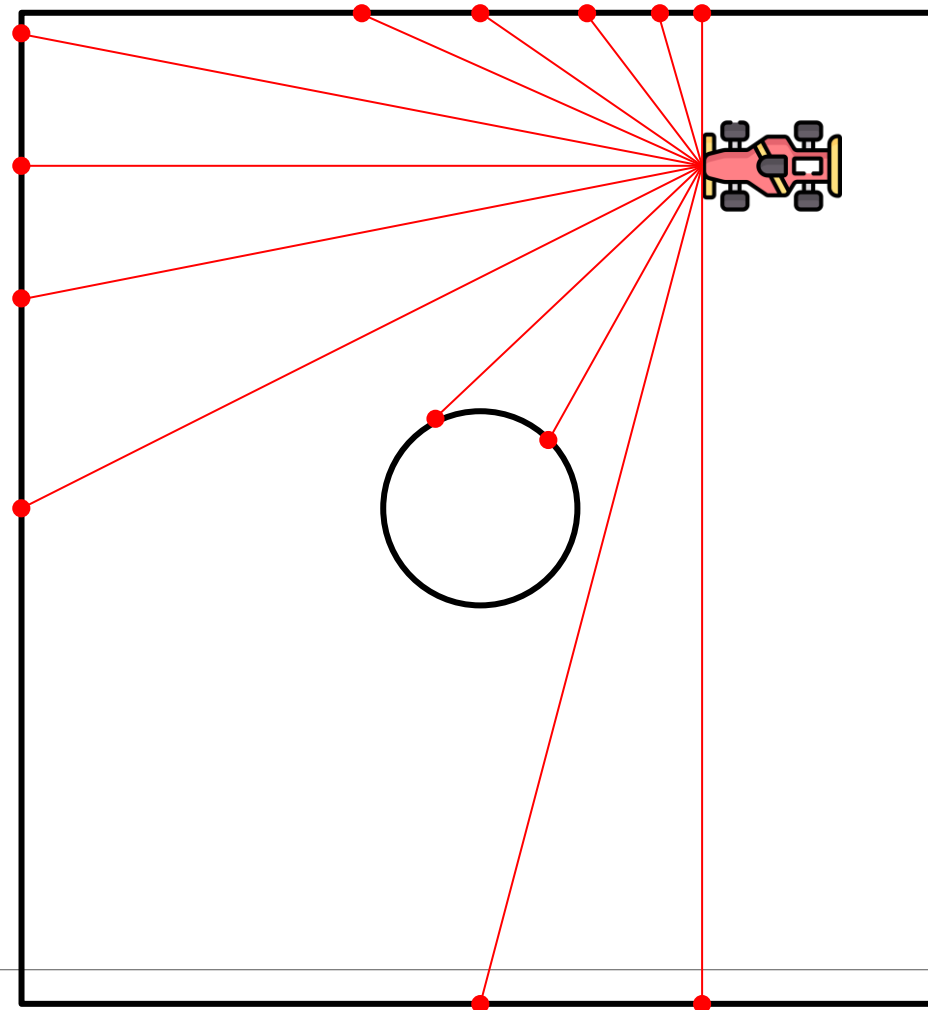
Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

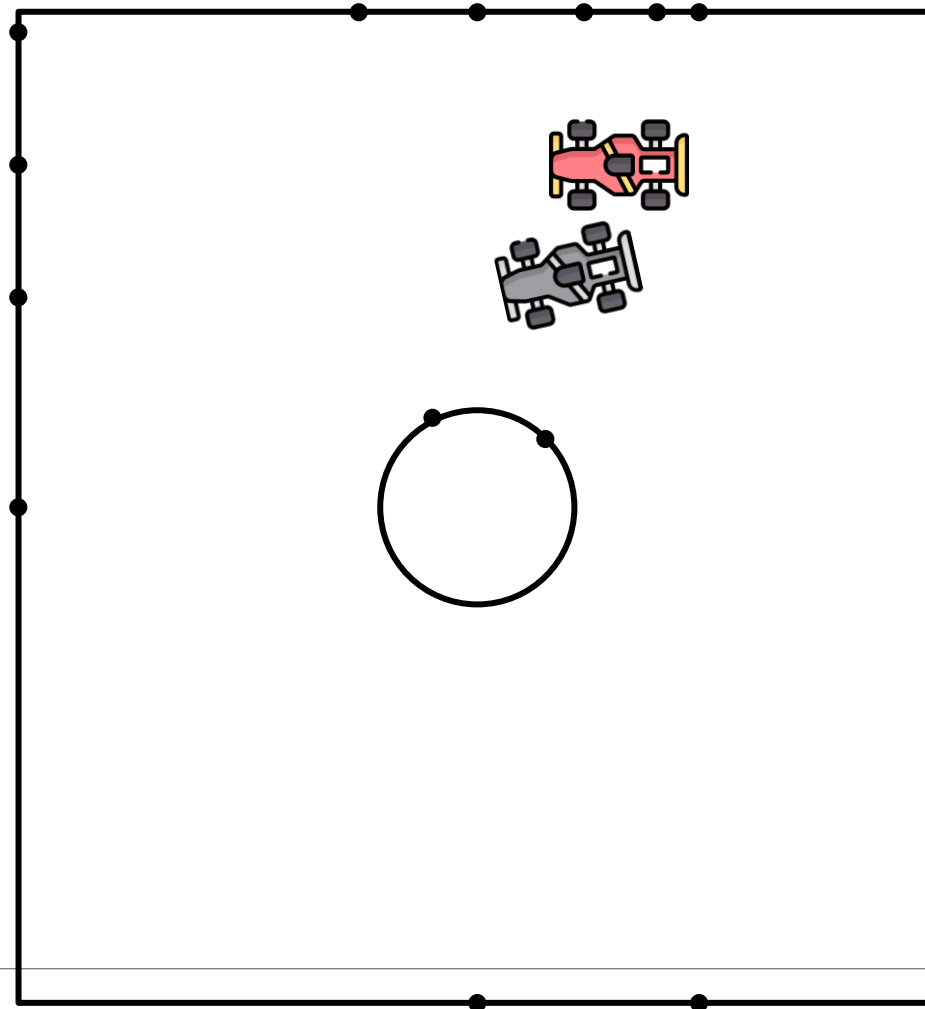
But in the real world...



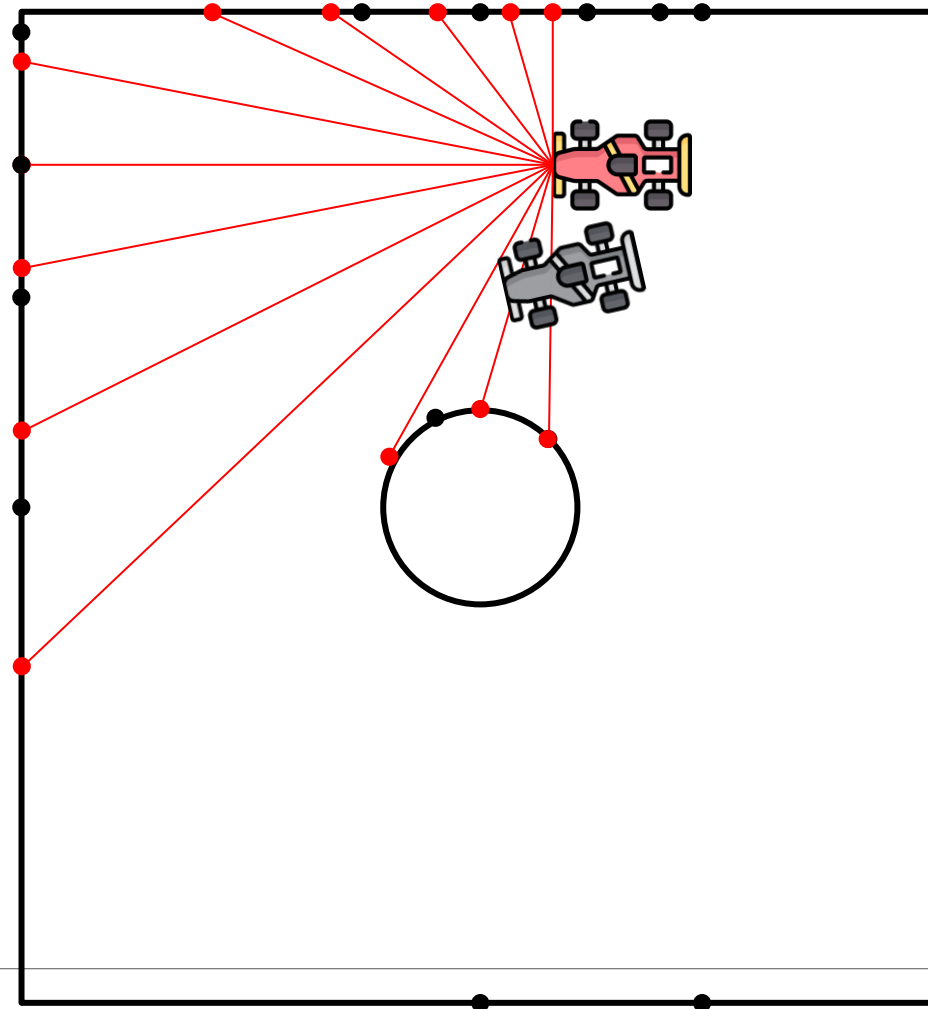
But in the real world...



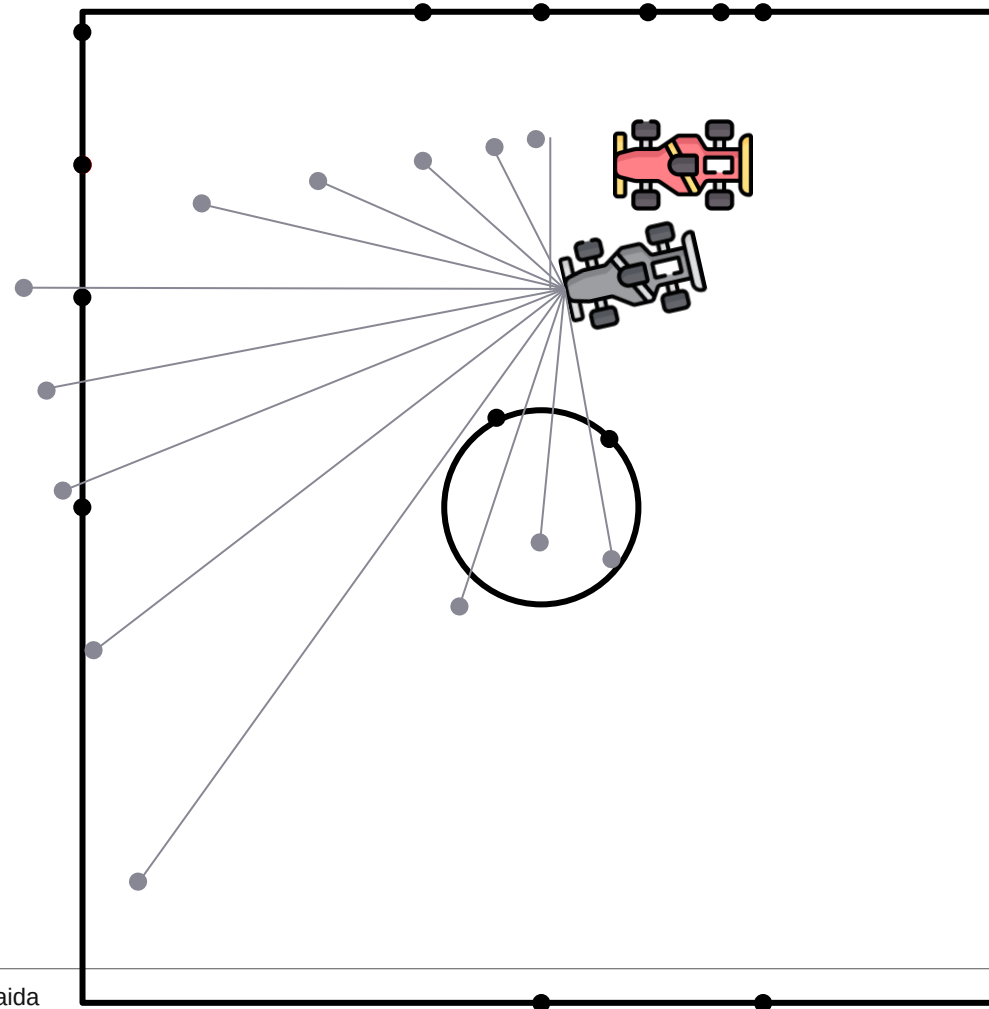
But in the real world...



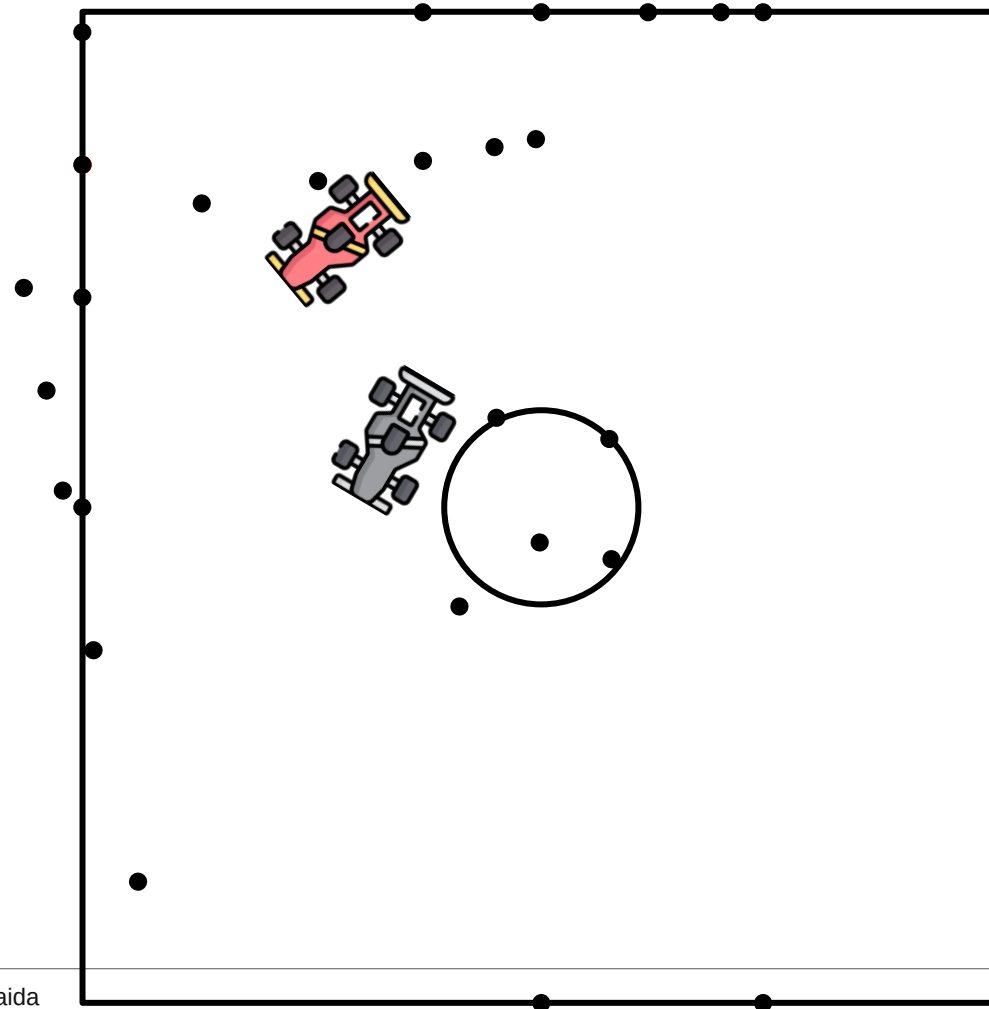
But in the real world...



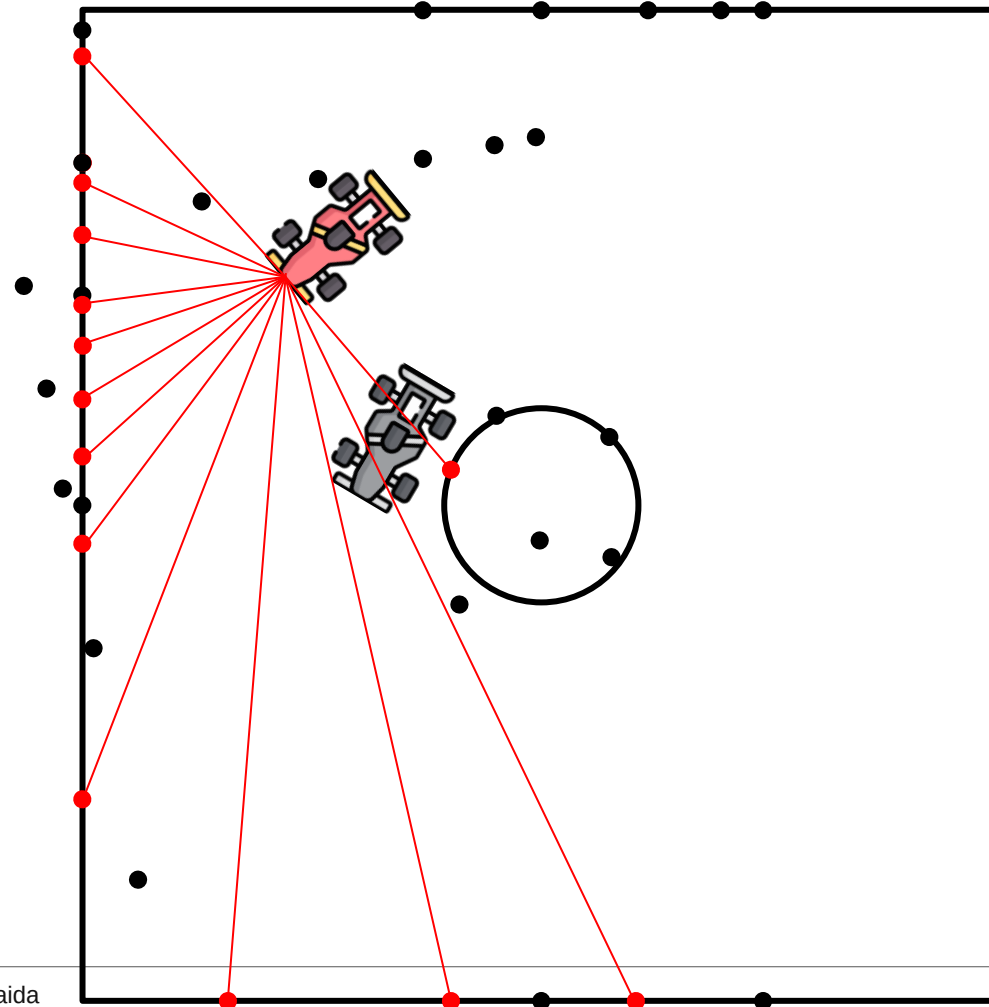
But in the real world...



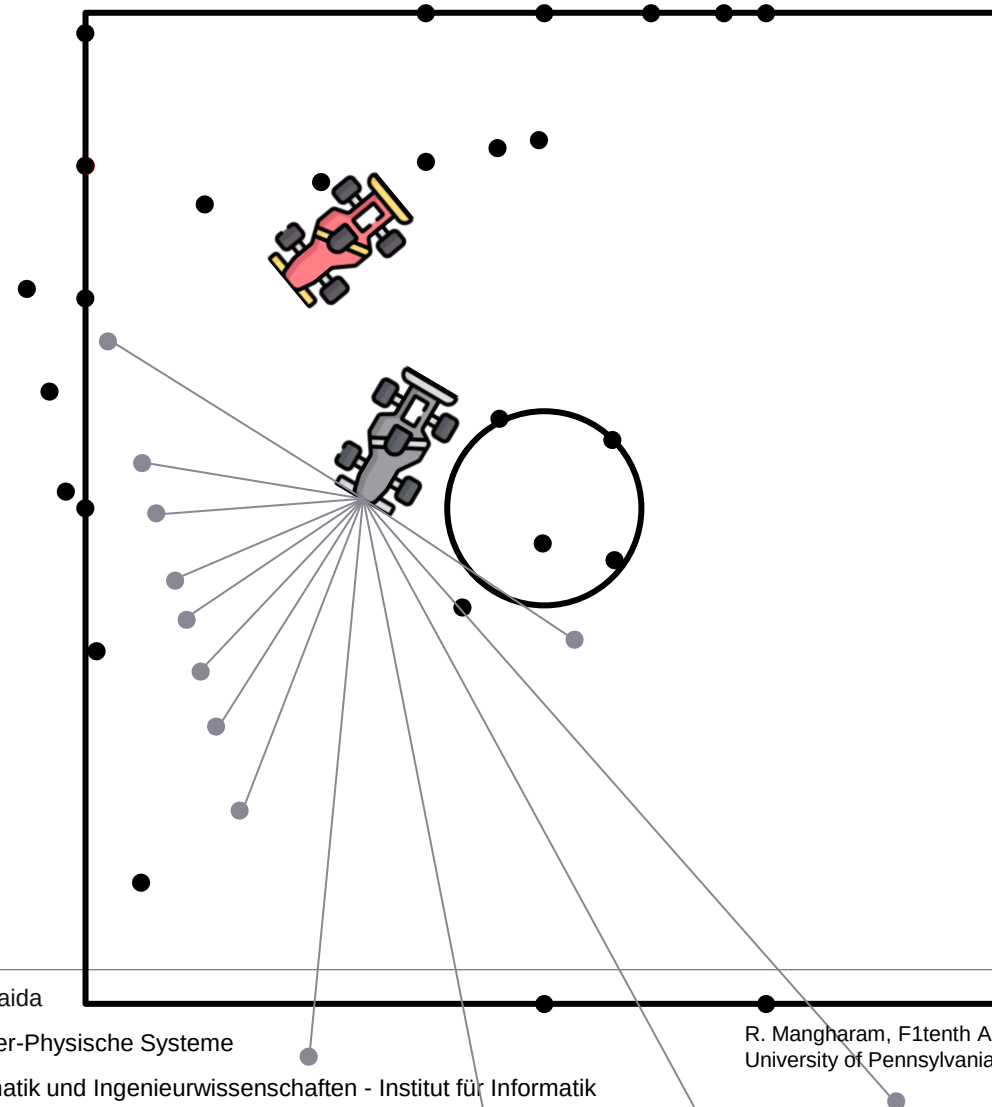
But in the real world...



But in the real world...



But in the real world...



13.10.2026

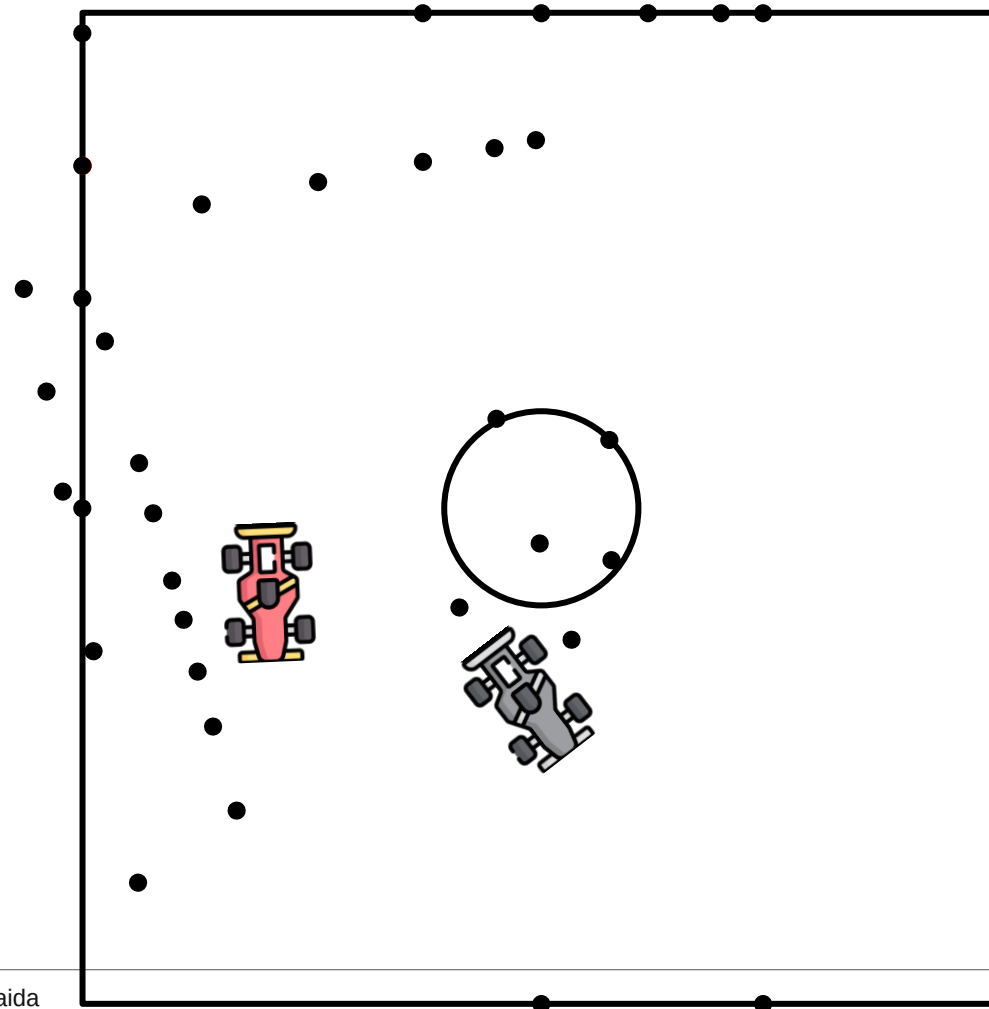
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

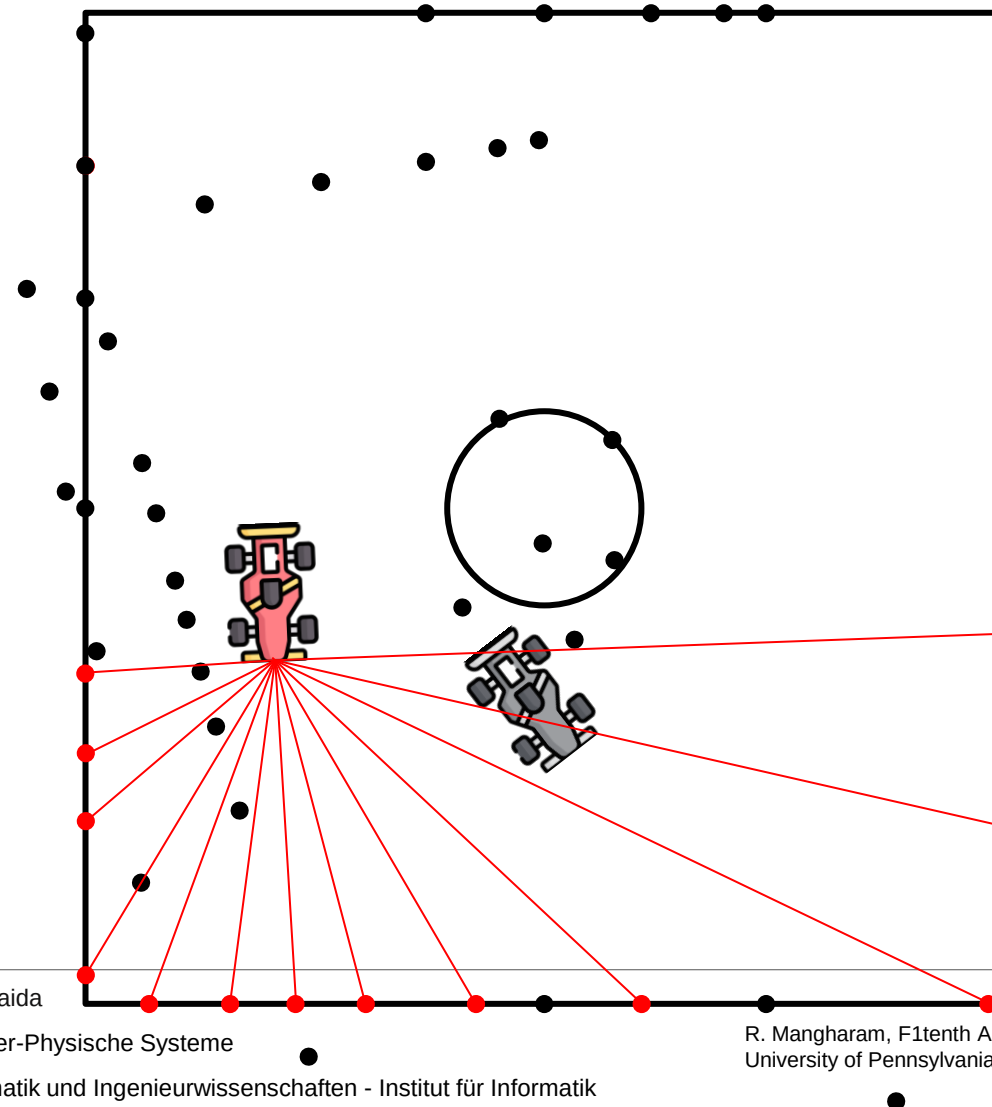
Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

But in the real world...



But in the real world...



13.10.2026

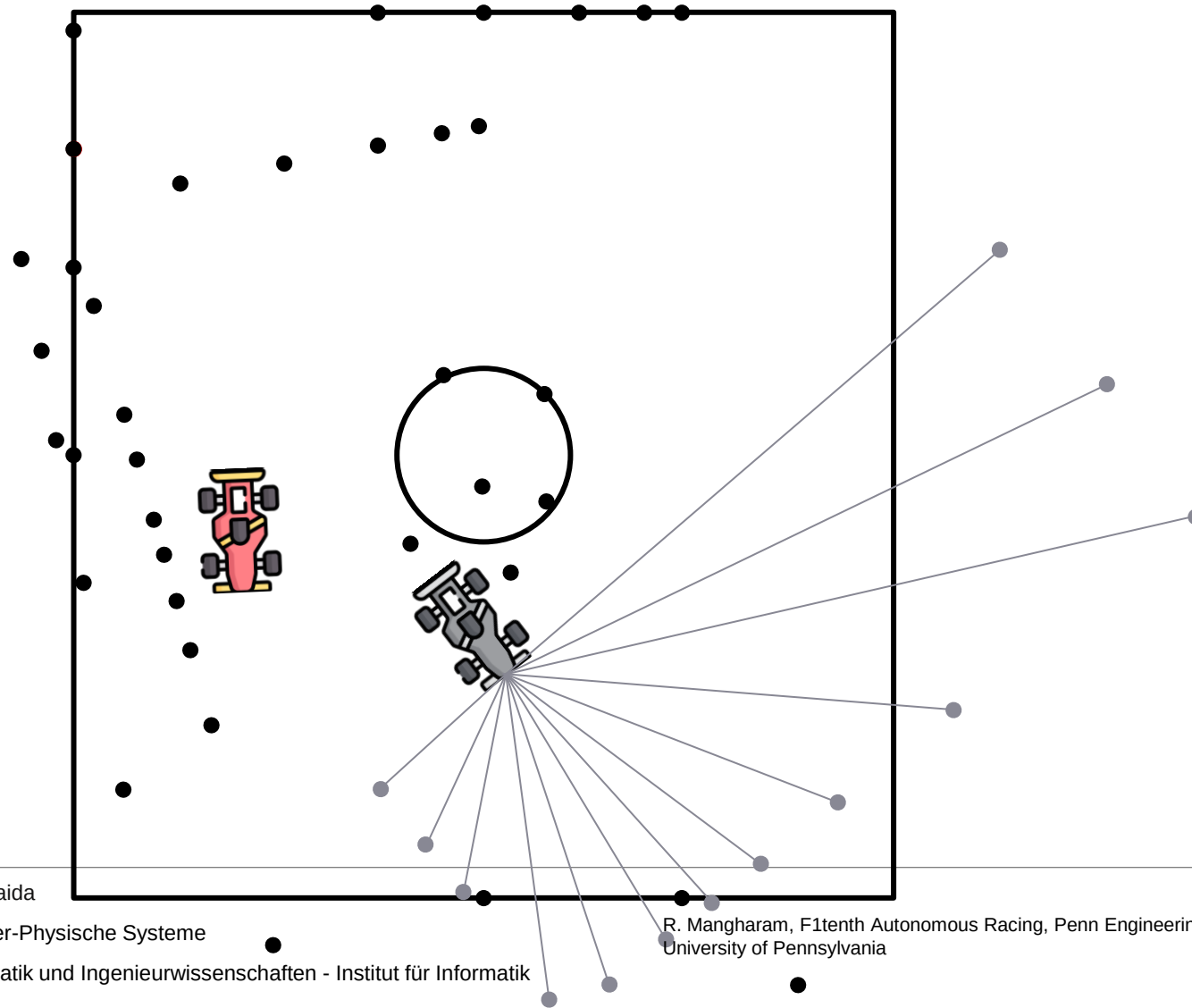
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

But in the real world...



13.10.2026

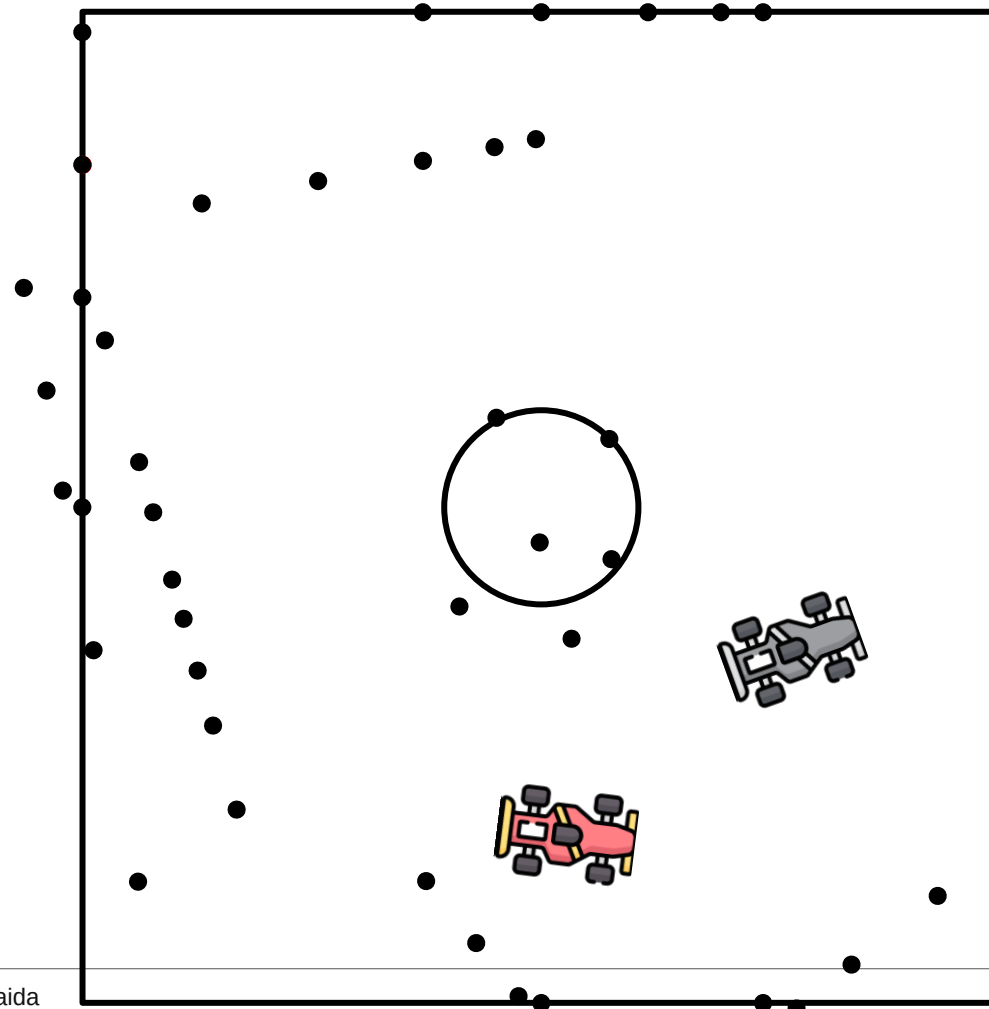
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

But in the real world...



13.10.2026

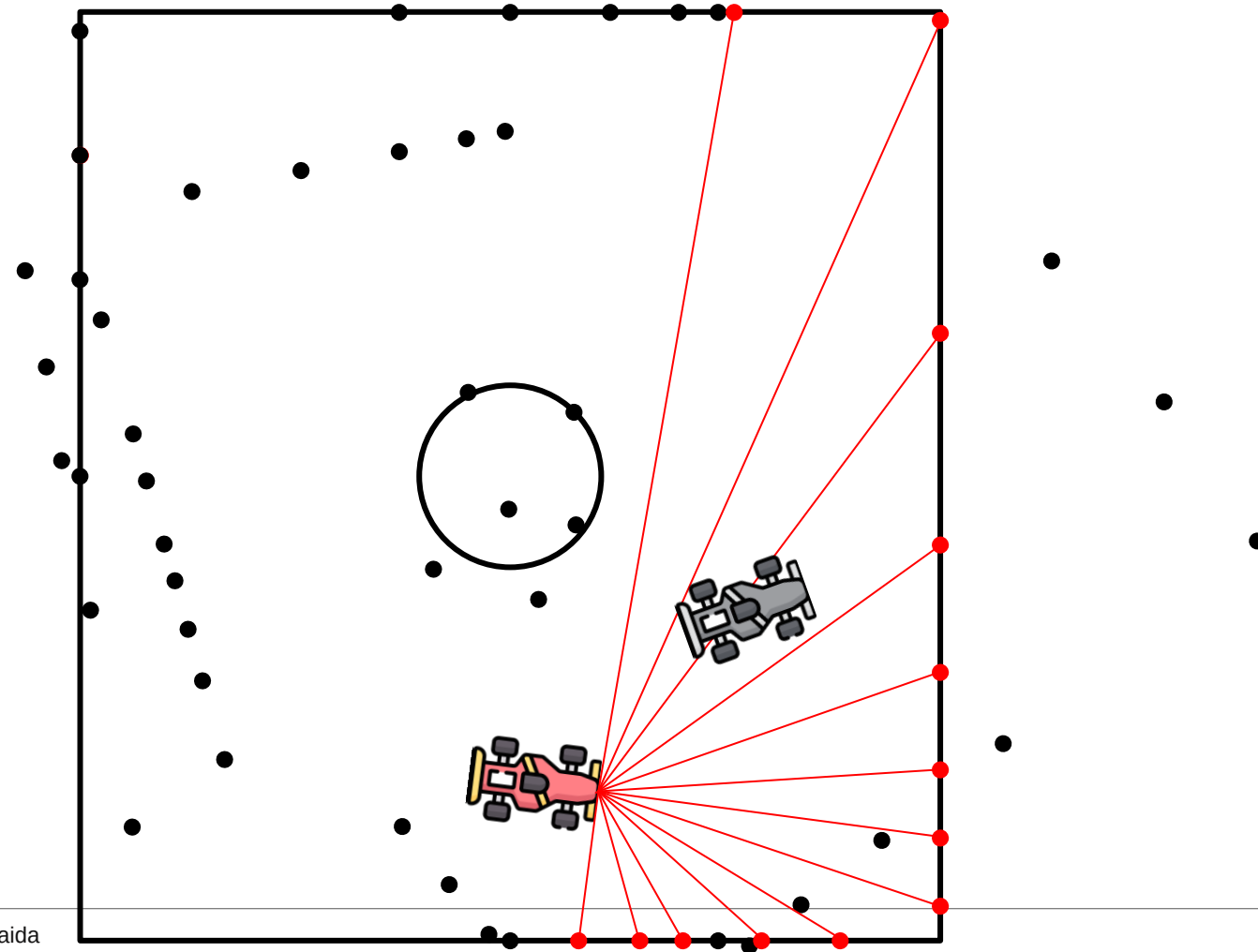
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

But in the real world...



13.10.2026

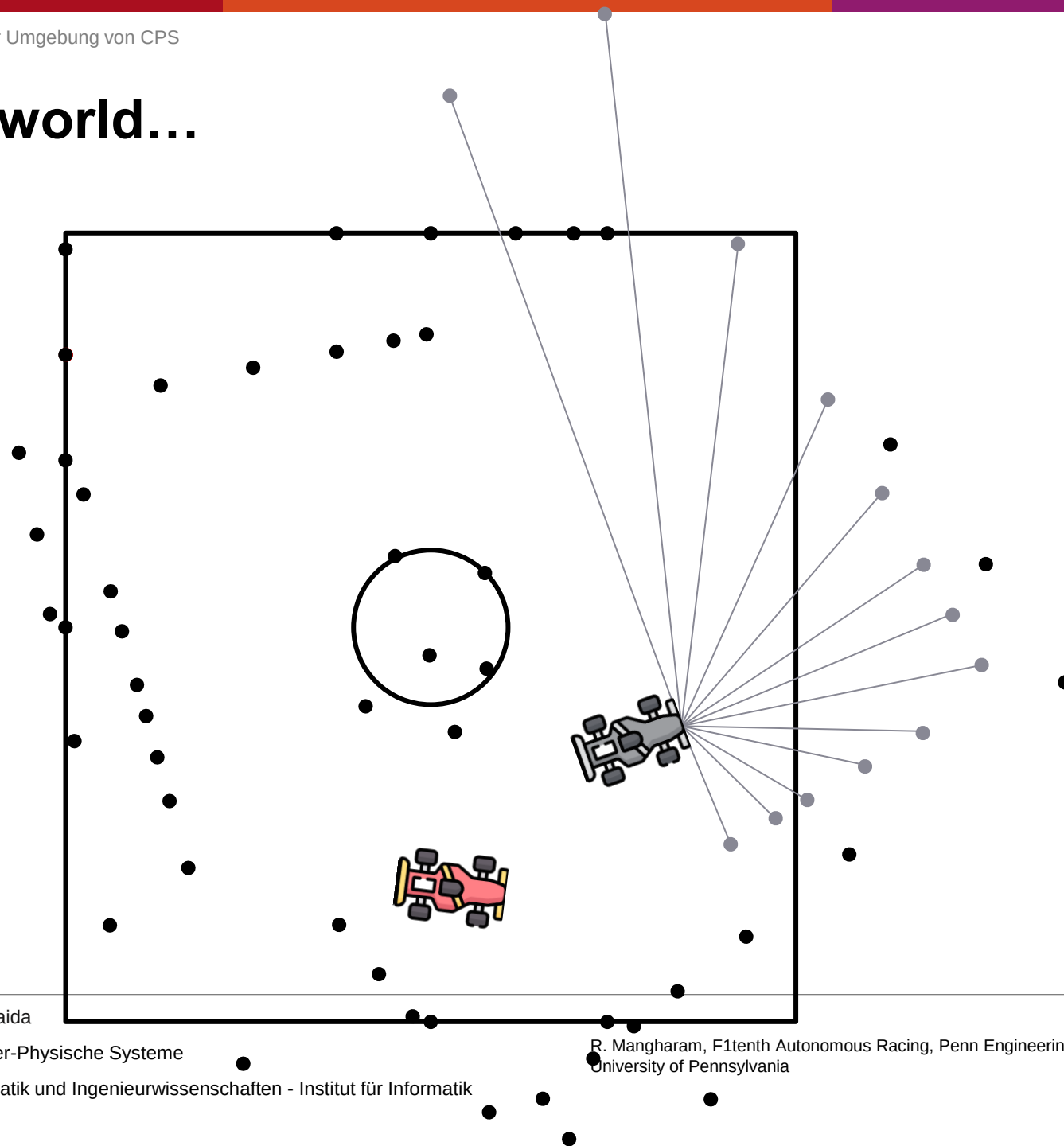
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

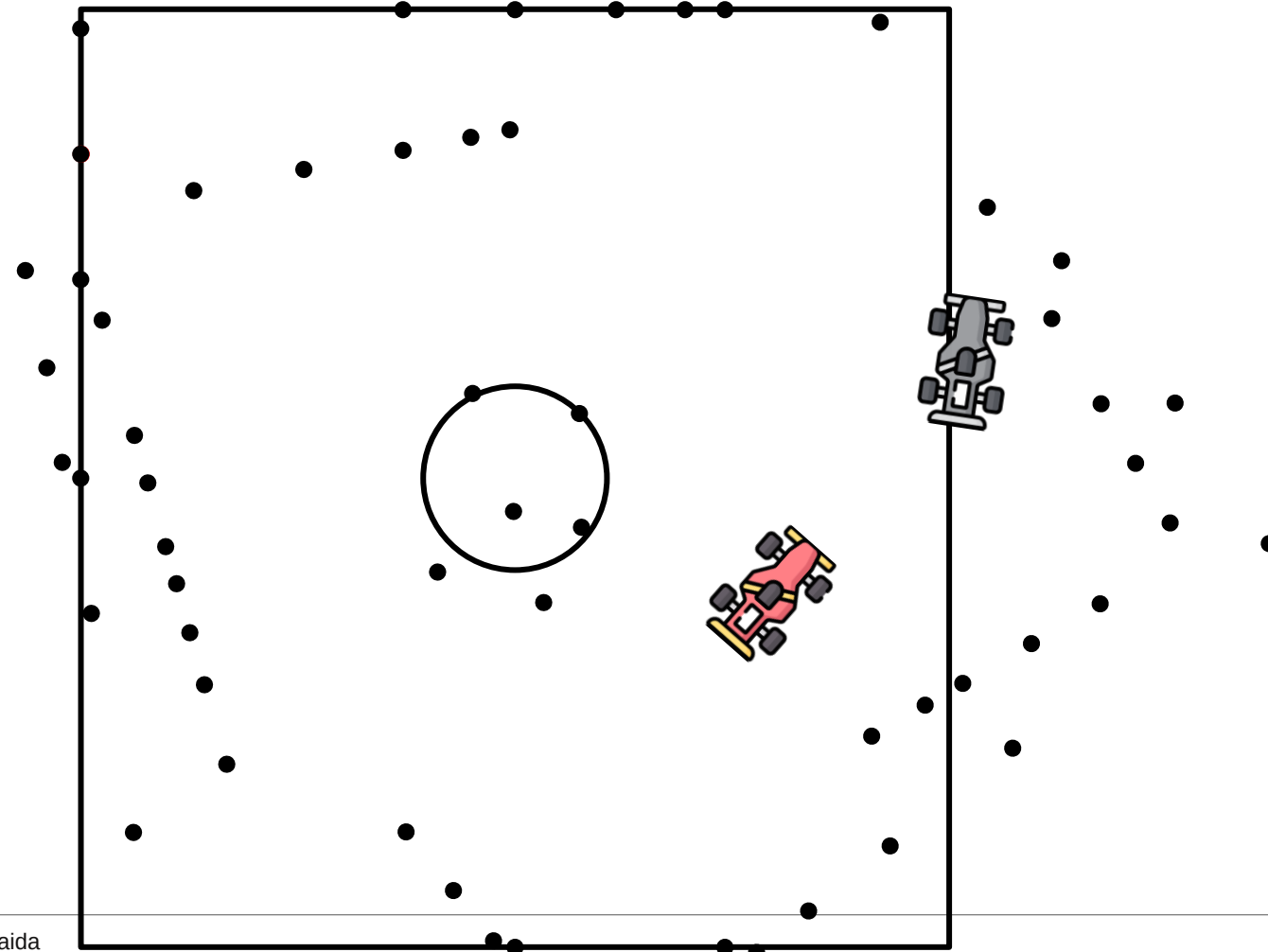
Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

But in the real world...



But in the real world...



13.10.2026

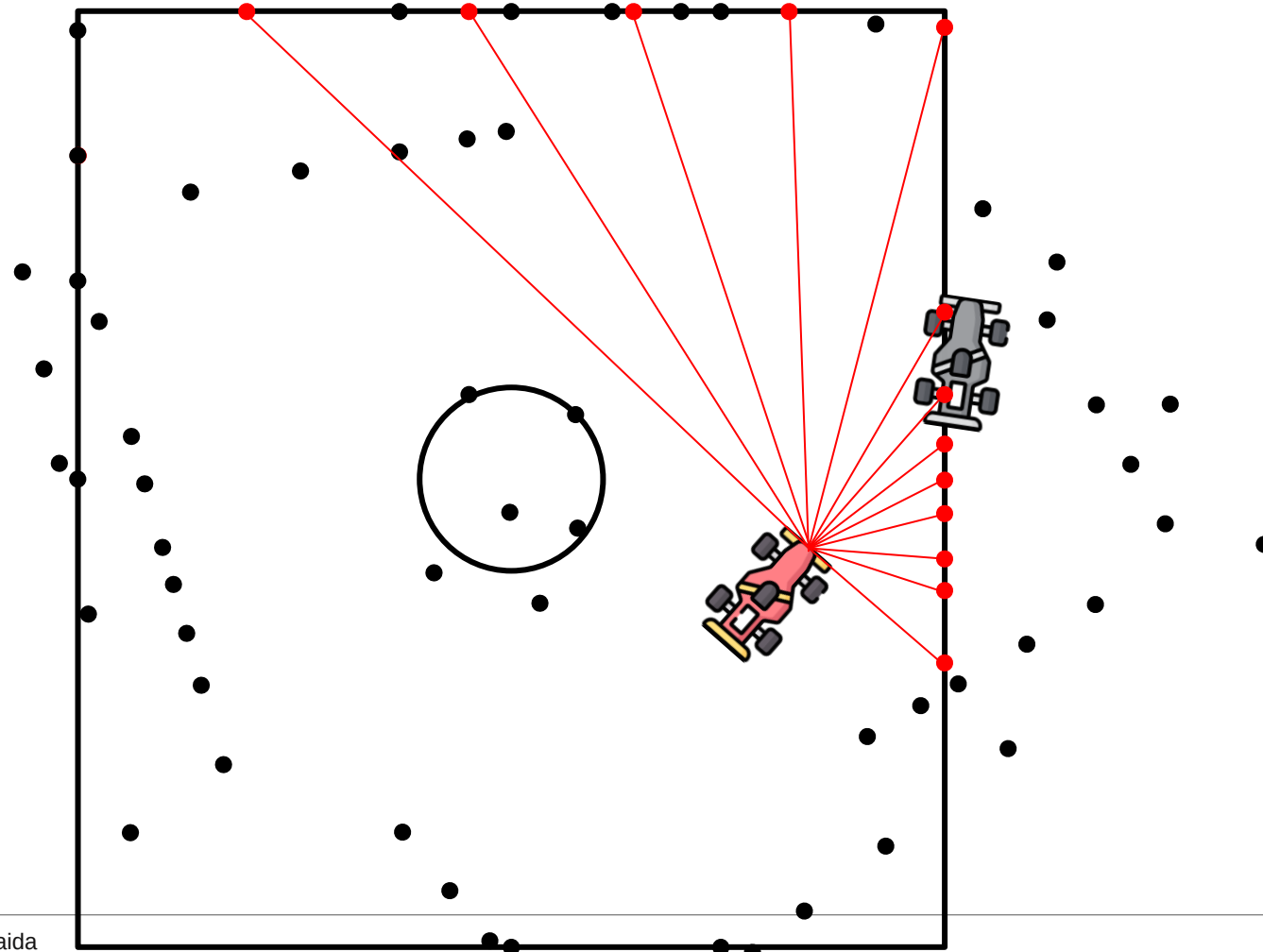
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

But in the real world...



13.10.2026

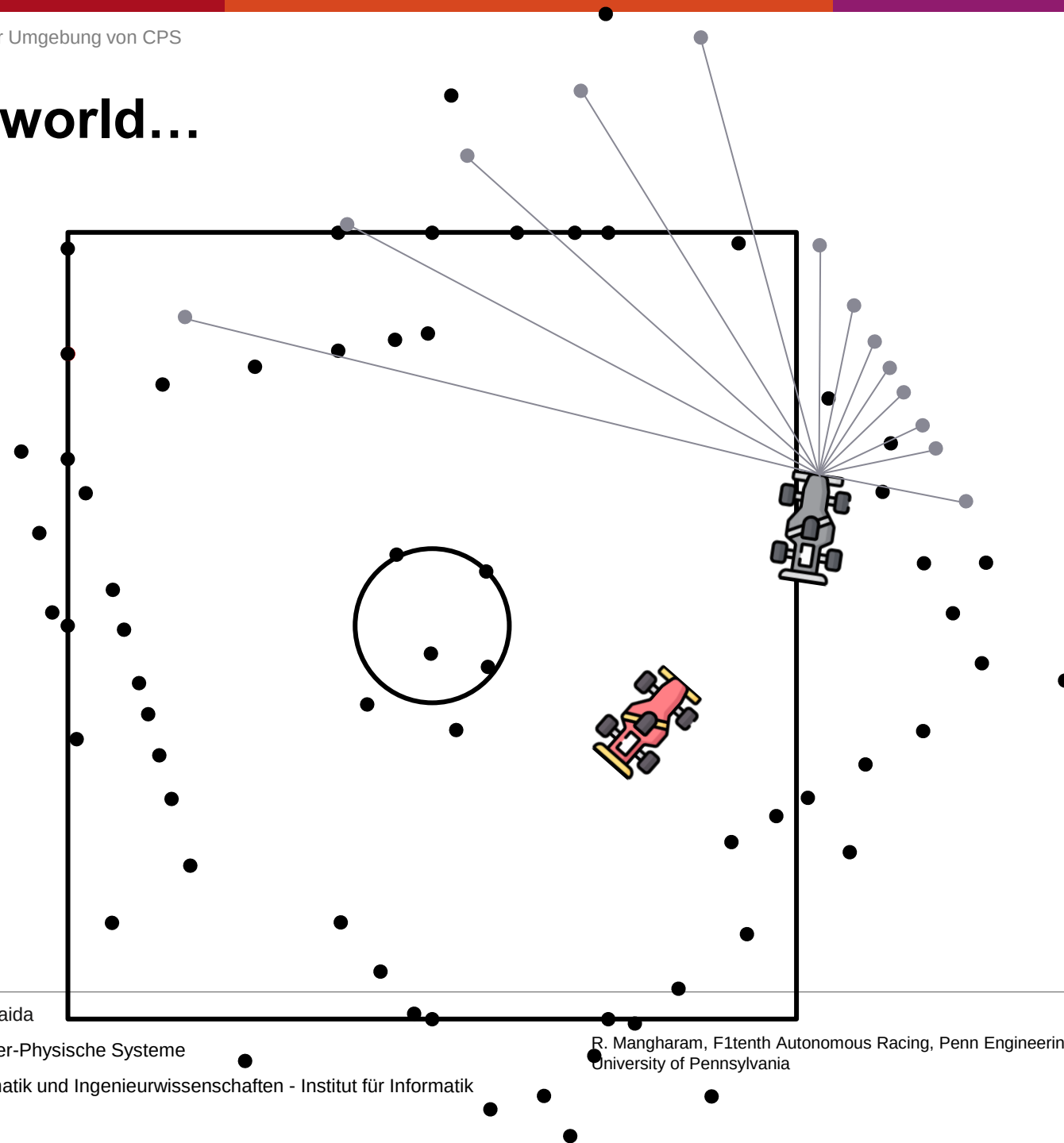
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

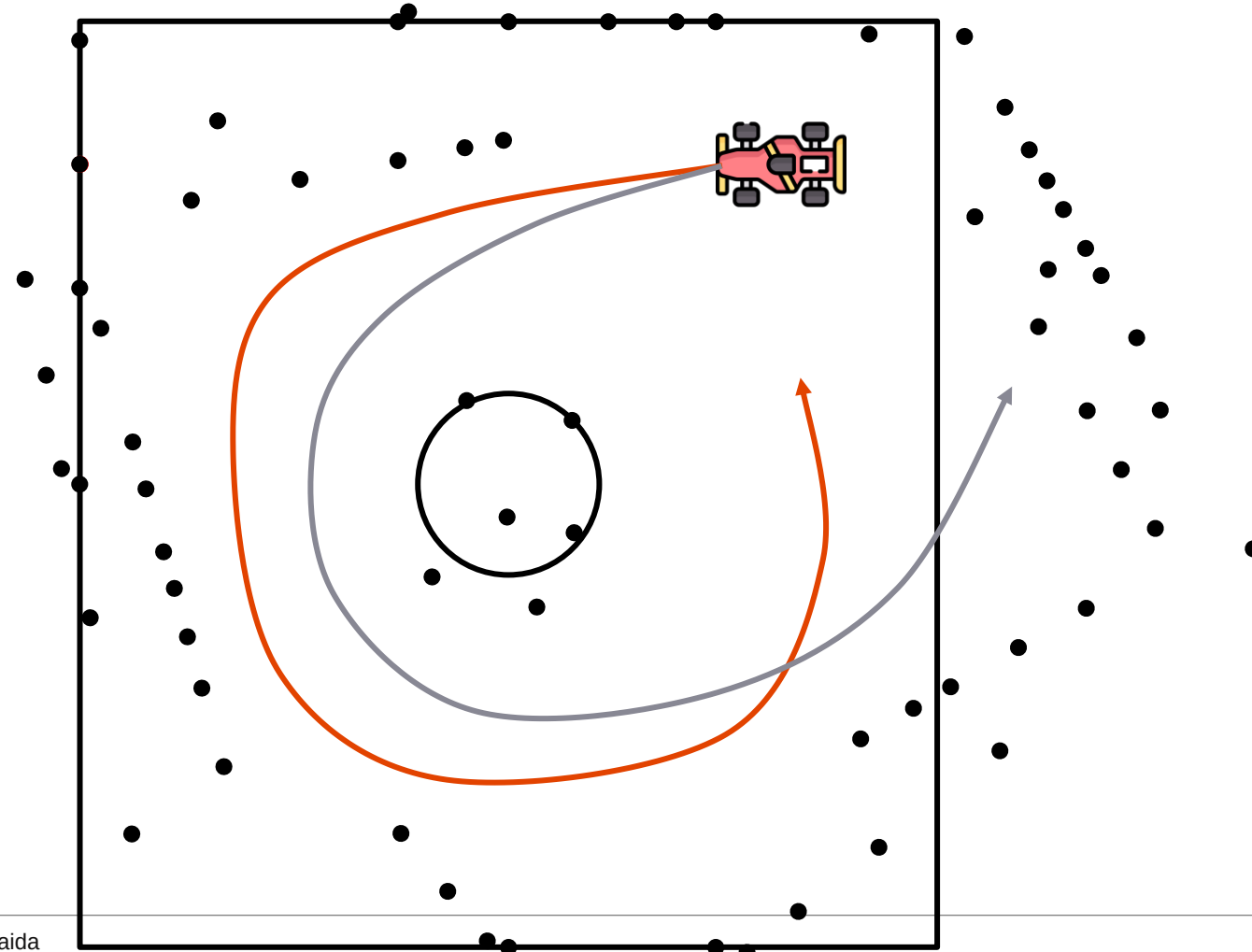
Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

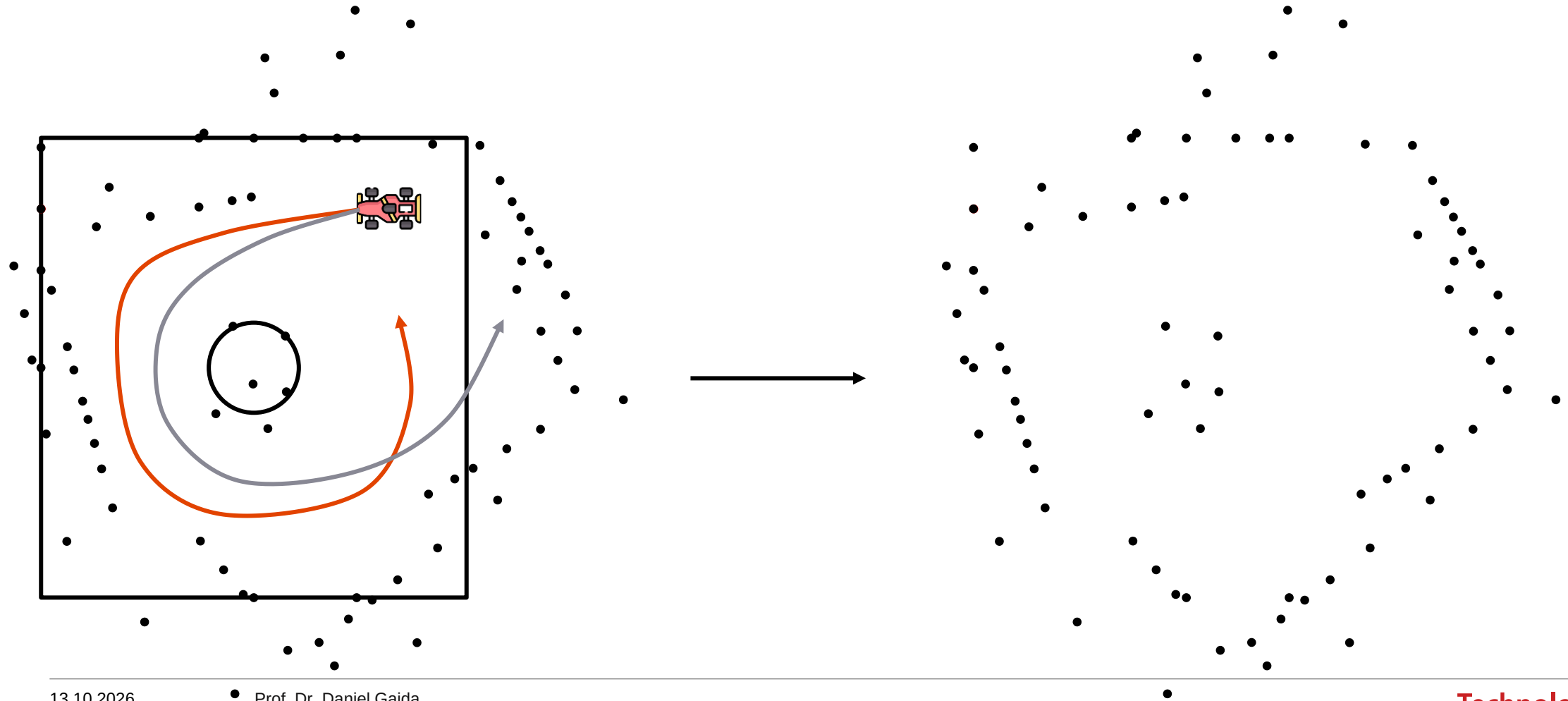
But in the real world...



But in the real world...



But in the real world...



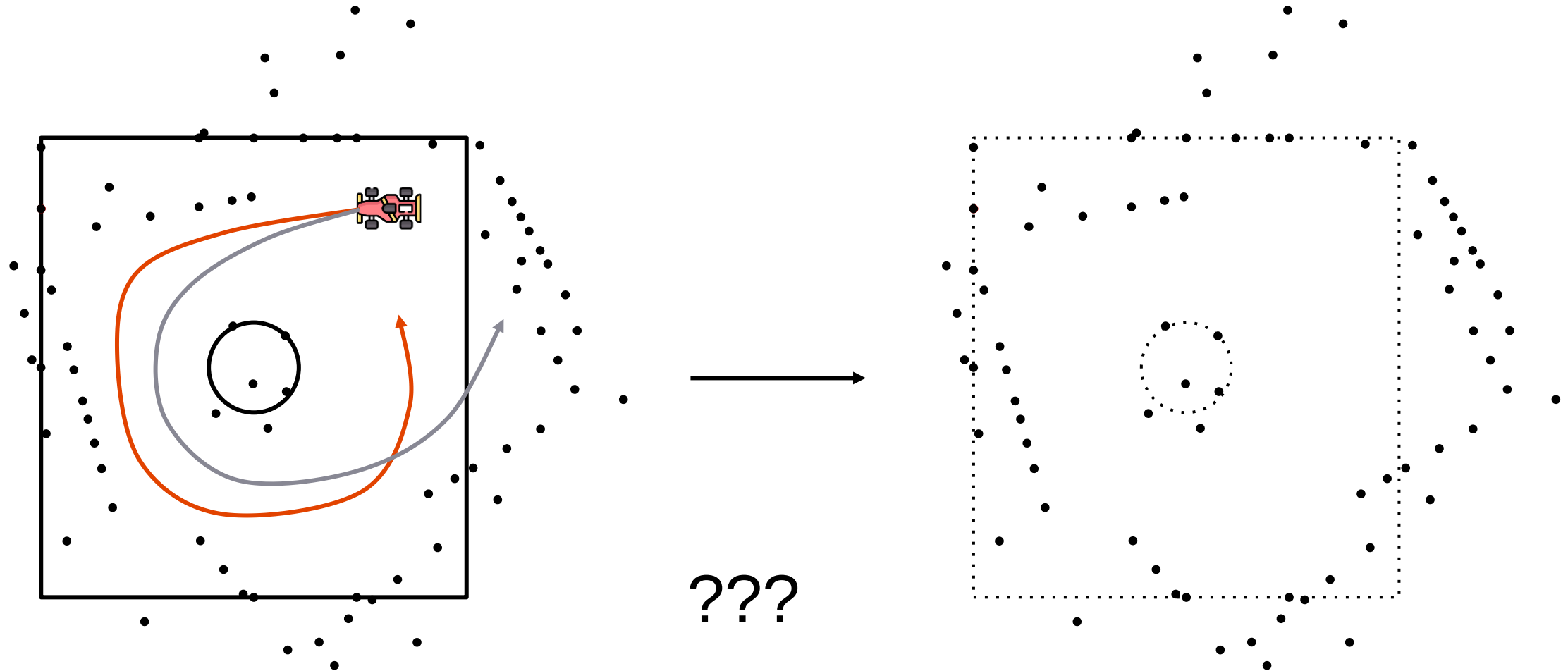
13.10.2026

• Prof. Dr. Daniel Gaida
Professor für Cyber-Physische Systeme
Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

But in the real world...



SLAM overview

Generally 2 different approaches:

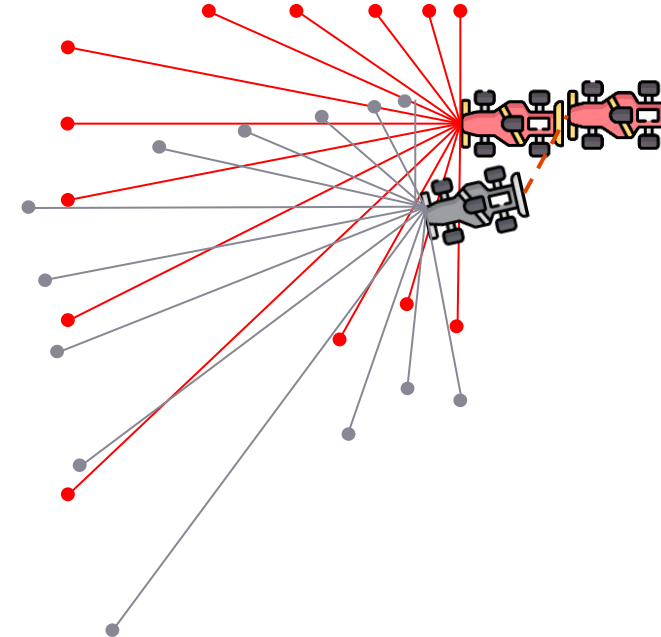
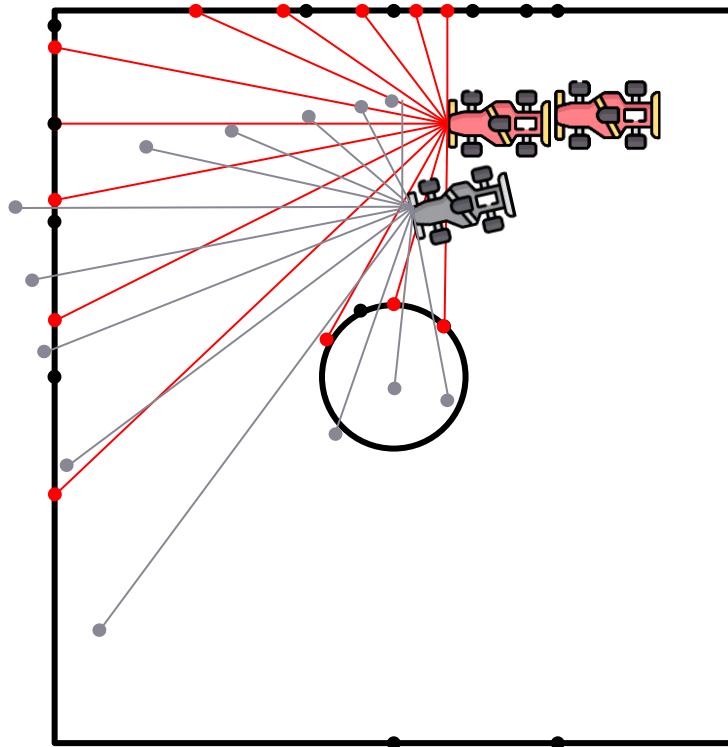
- Filtering-based
 - Use Extended Kalman Filter or Particle Filter for state estimation and build the map based on that estimation
 - Examples are GMapping, HectorSLAM
- -
 -
 -
 -

SLAM overview

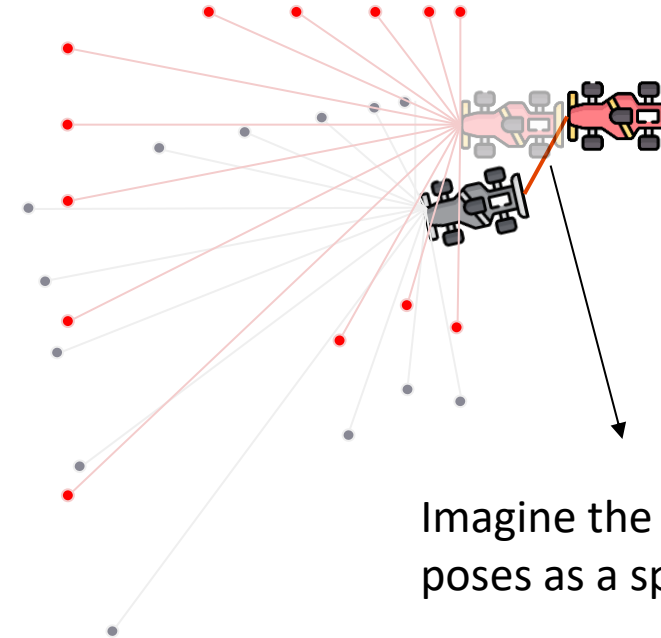
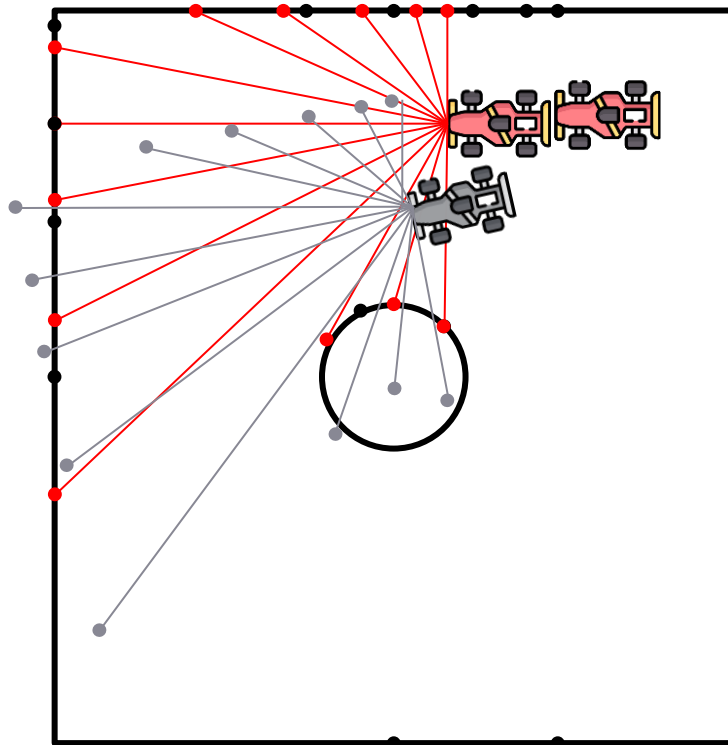
Generally 2 different approaches:

- Filtering-based
 - Use Extended Kalman Filter or Particle Filter for state estimation and build the map based on that estimation
 - Examples are GMapping, HectorSLAM
- Graph-based
 - Use pose graphs to store constraints between pose estimations
 - Use optimization for loop-closure to correct for pose estimations
 - Examples are KartoSLAM and Cartographer
 - We'll be using slam_toolbox, also graph-based

Pose Graphs: Edge Constraints

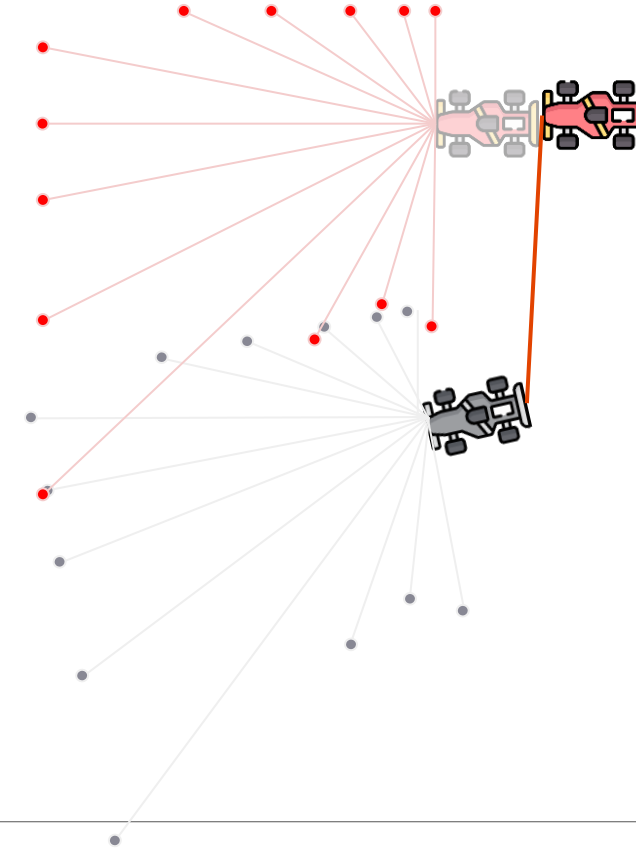
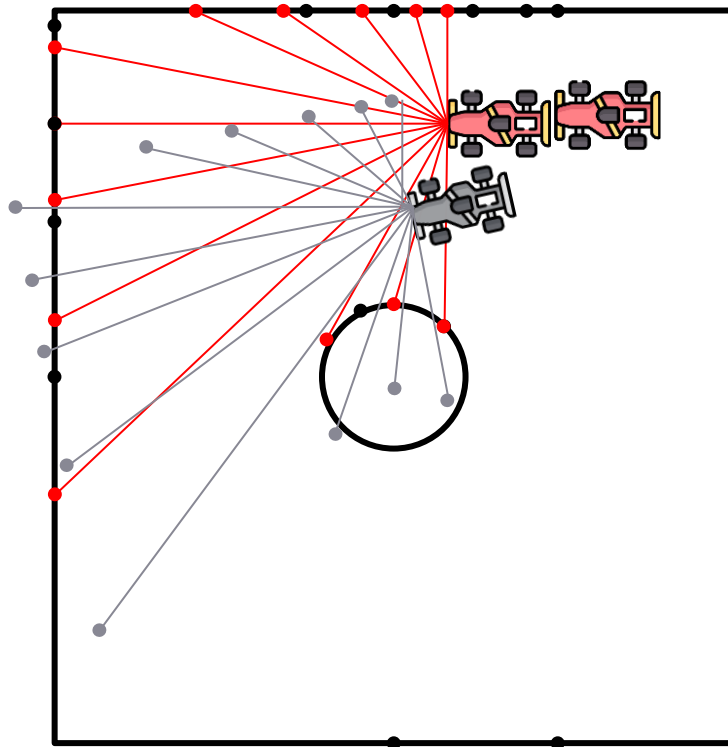


Pose Graphs: Edge Constraints

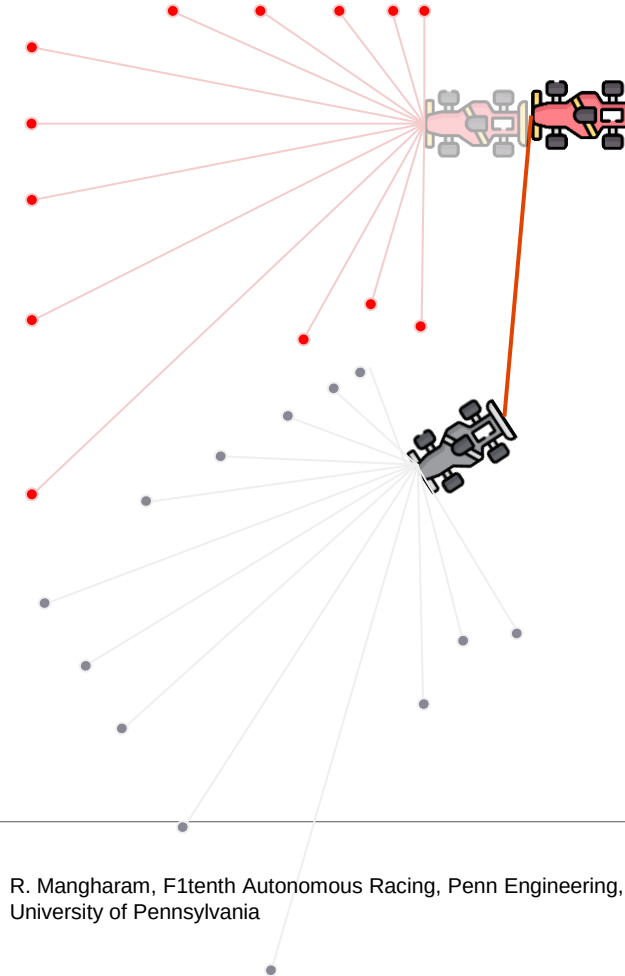
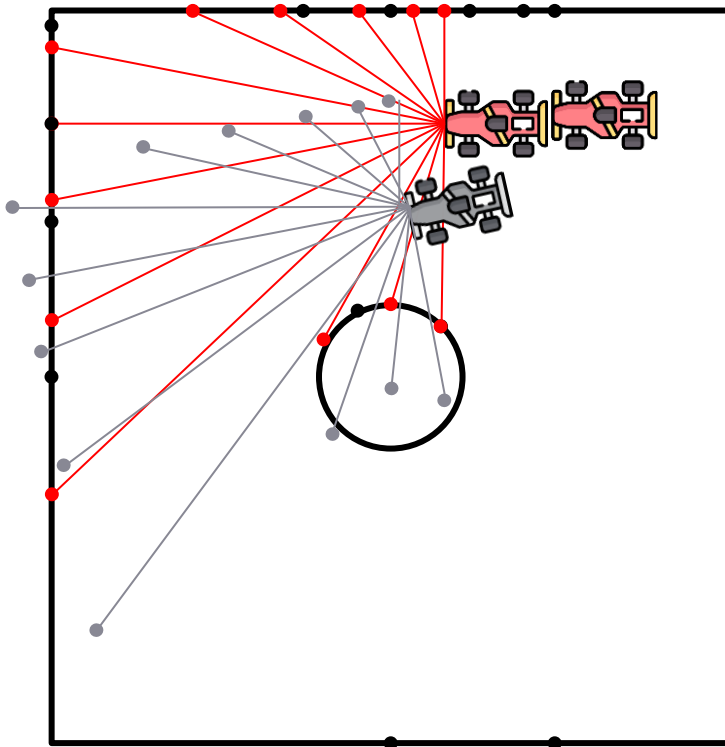


Imagine the constraint between poses as a spring.

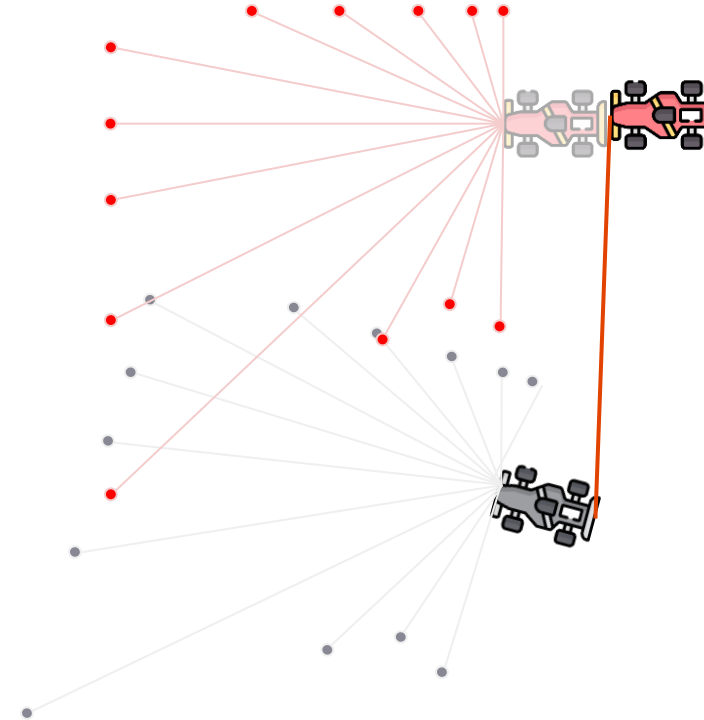
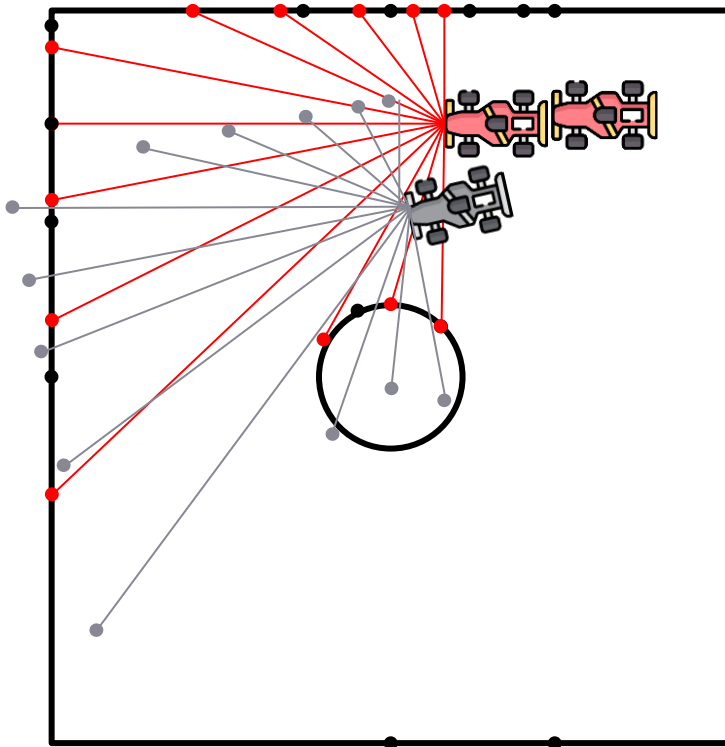
Pose Graphs: Edge Constraints



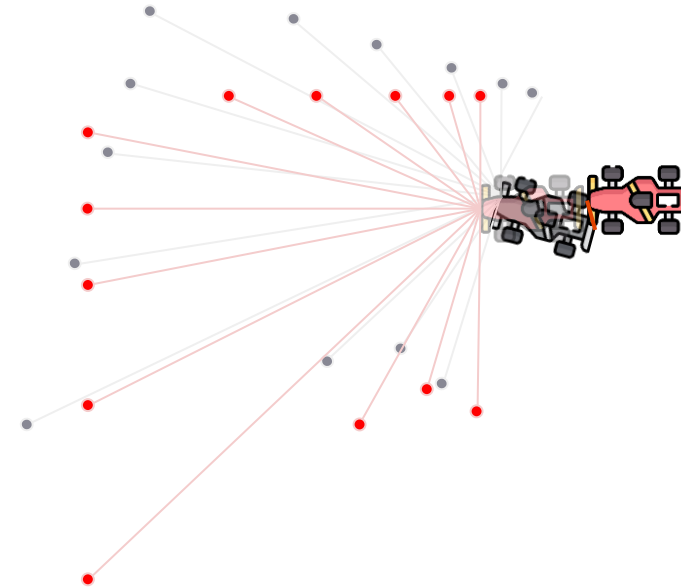
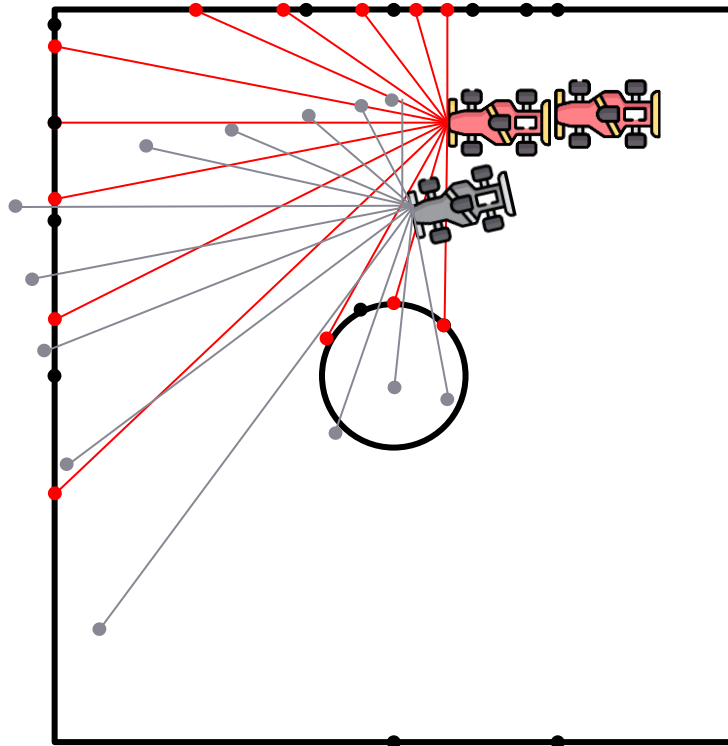
Pose Graphs: Edge Constraints



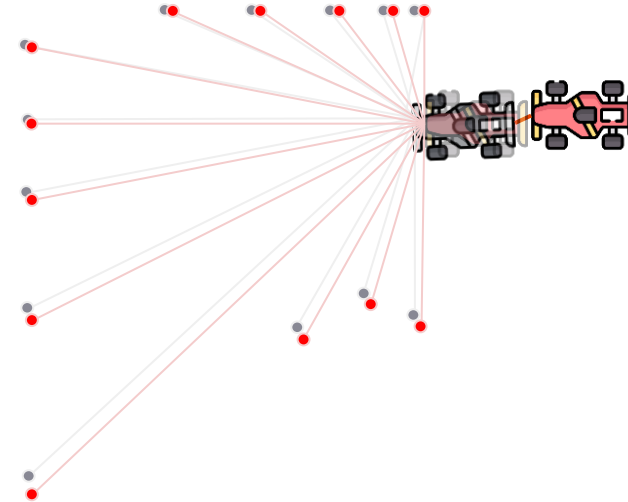
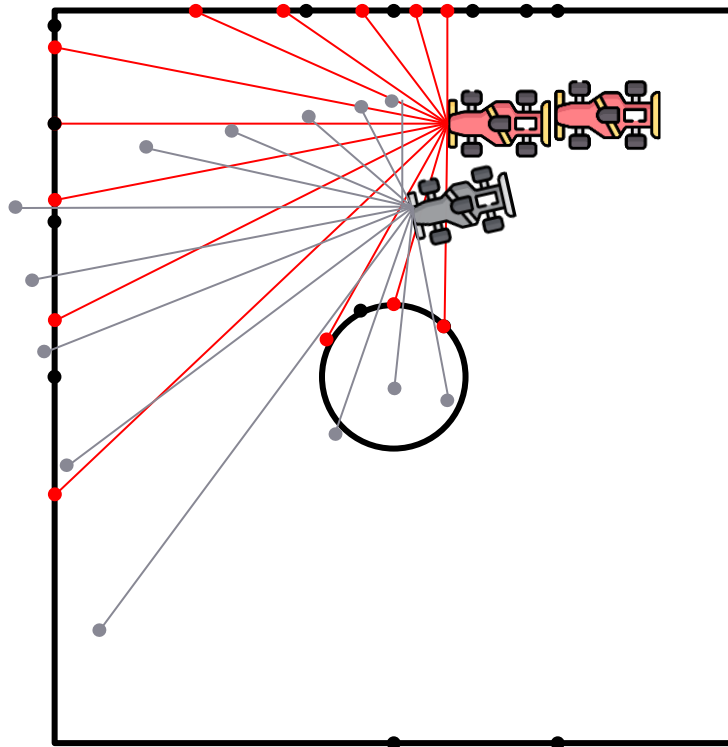
Pose Graphs: Edge Constraints



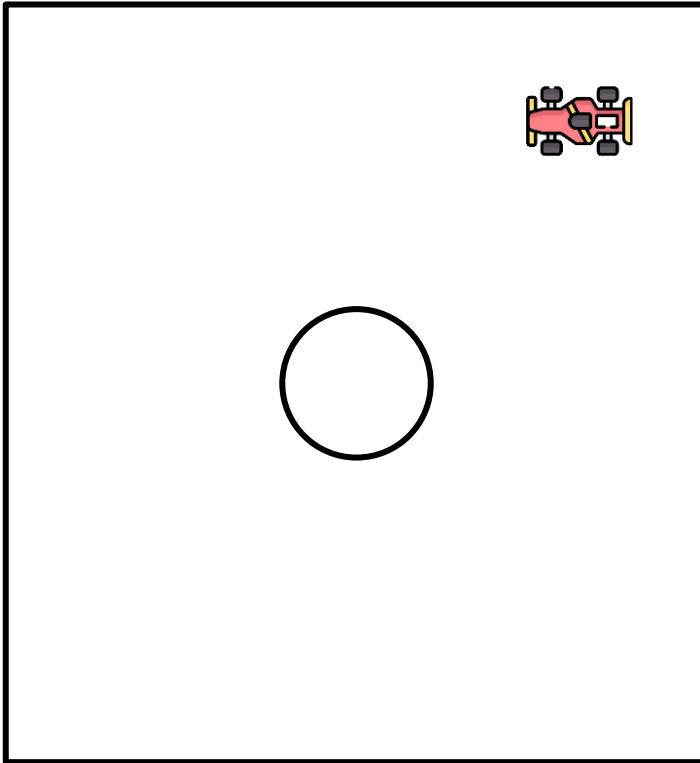
Pose Graphs: Edge Constraints



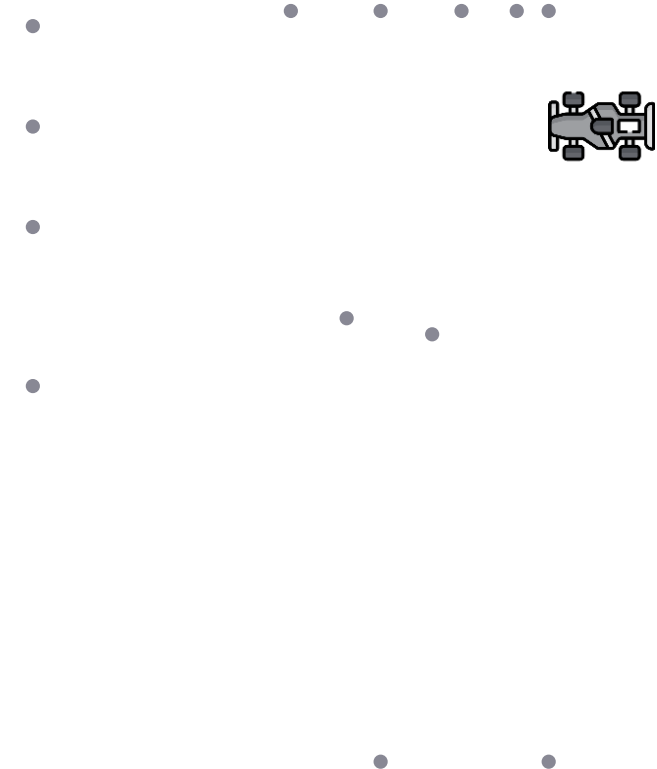
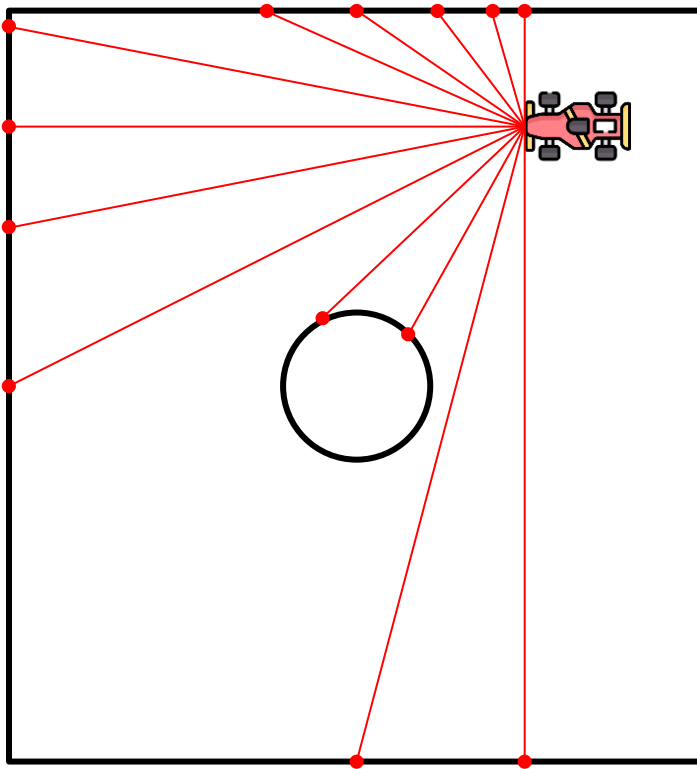
Pose Graphs: Edge Constraints



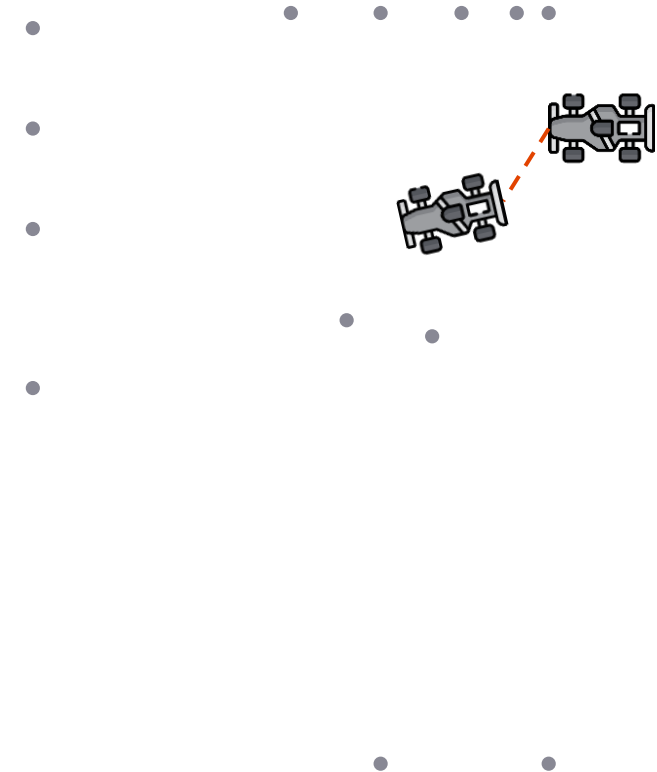
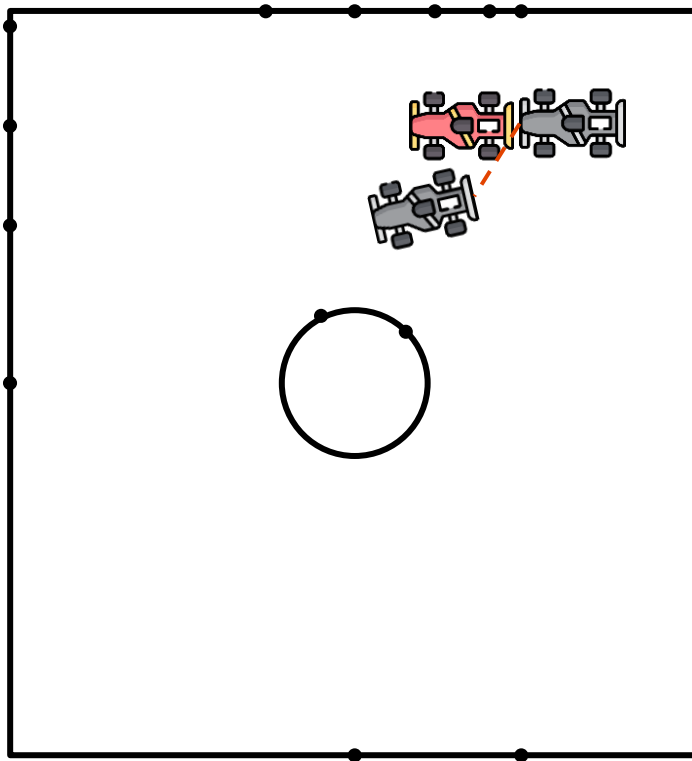
Pose Graphs



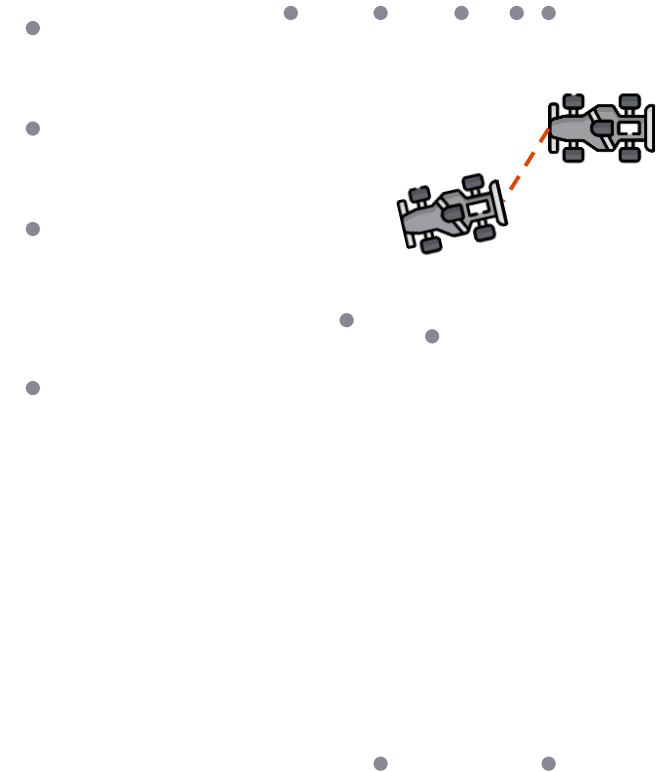
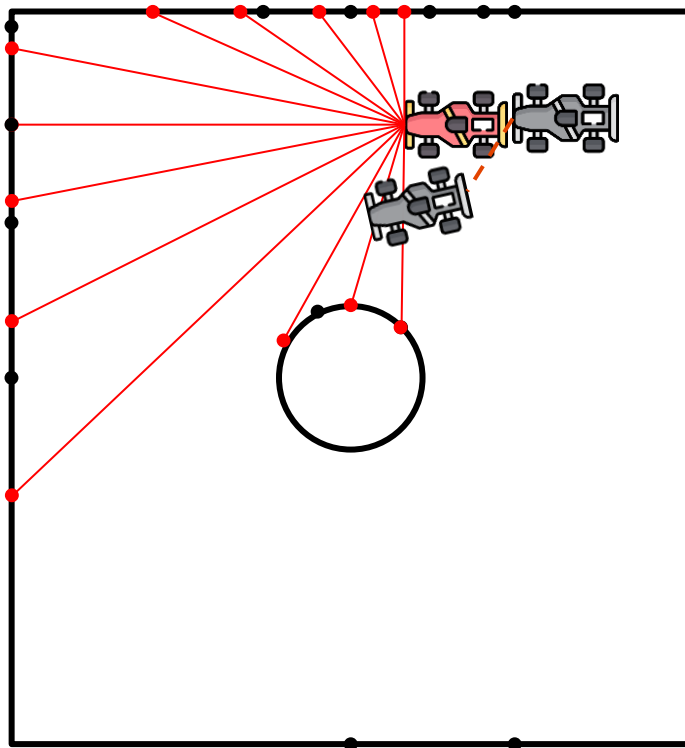
Pose Graphs



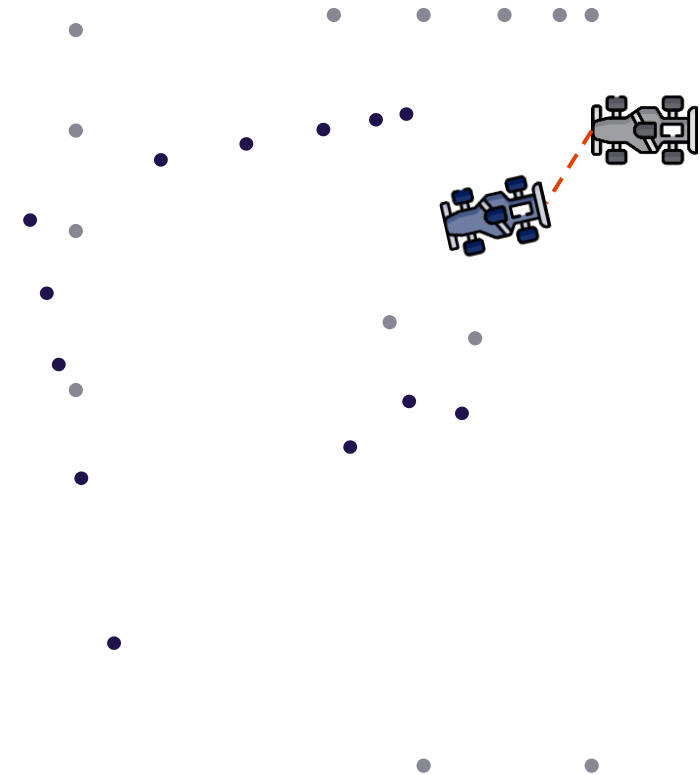
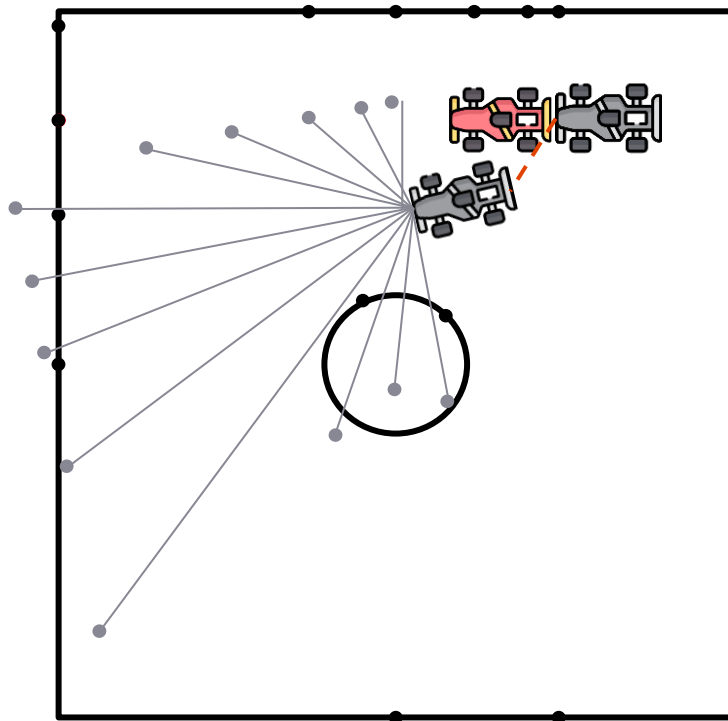
Pose Graphs



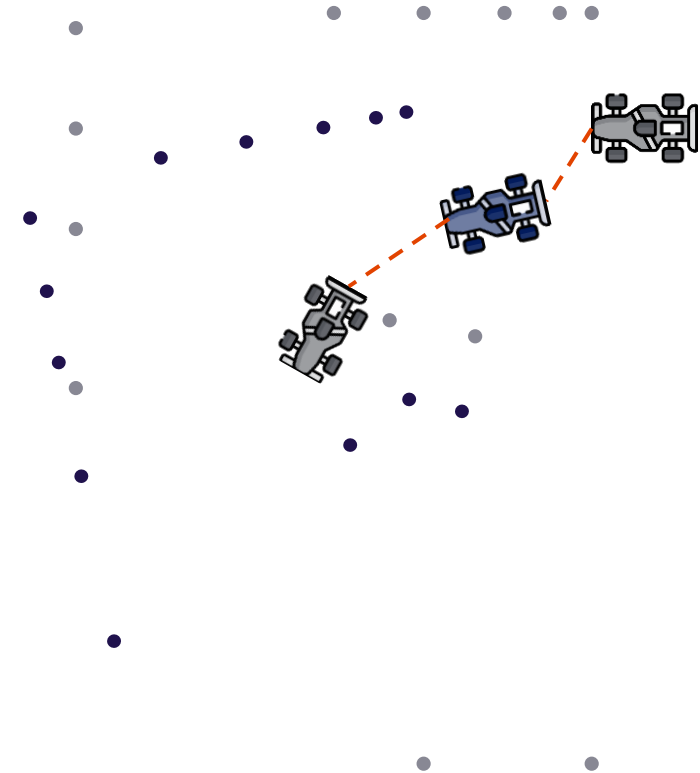
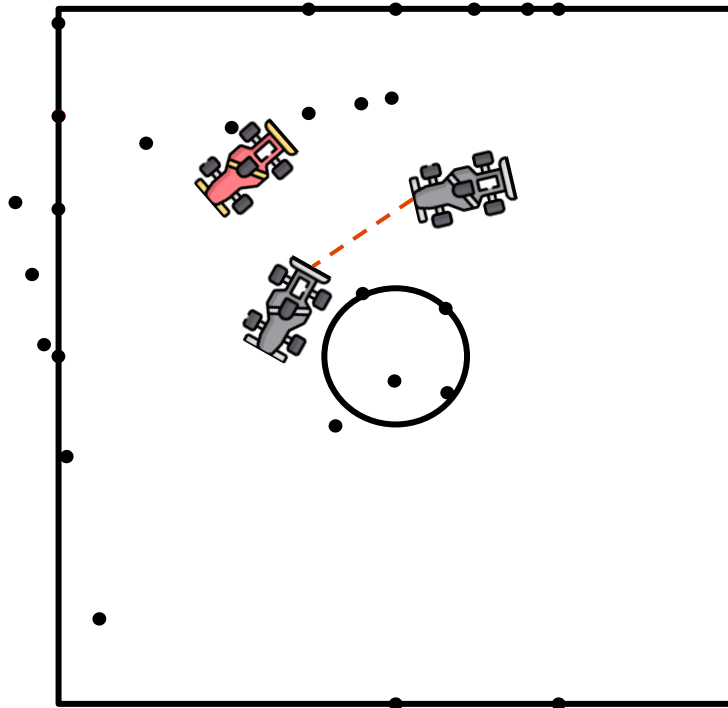
Pose Graphs



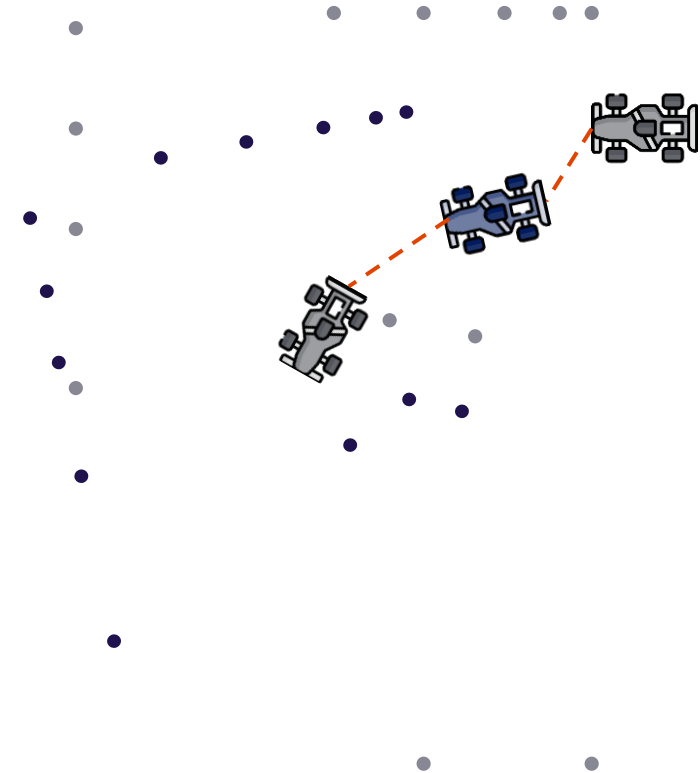
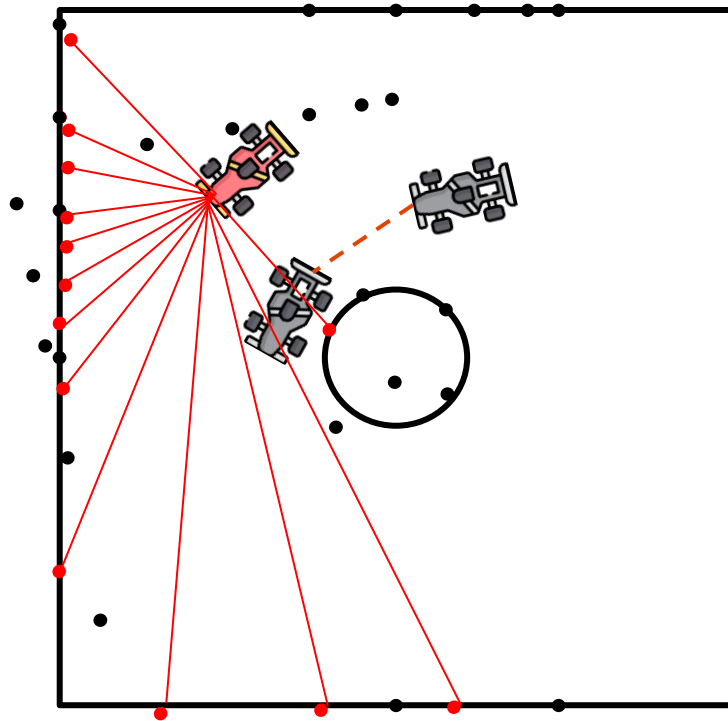
Pose Graphs



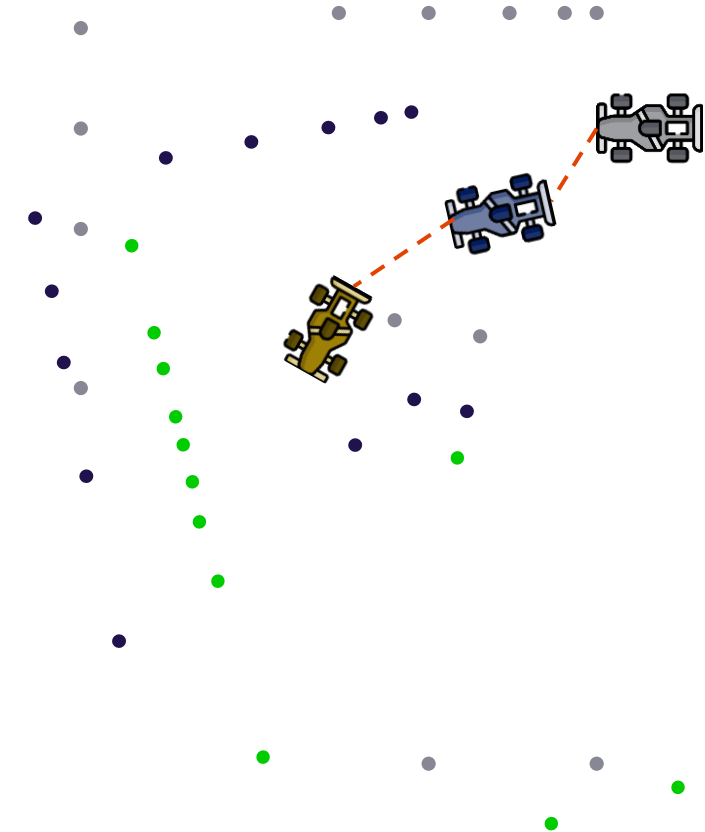
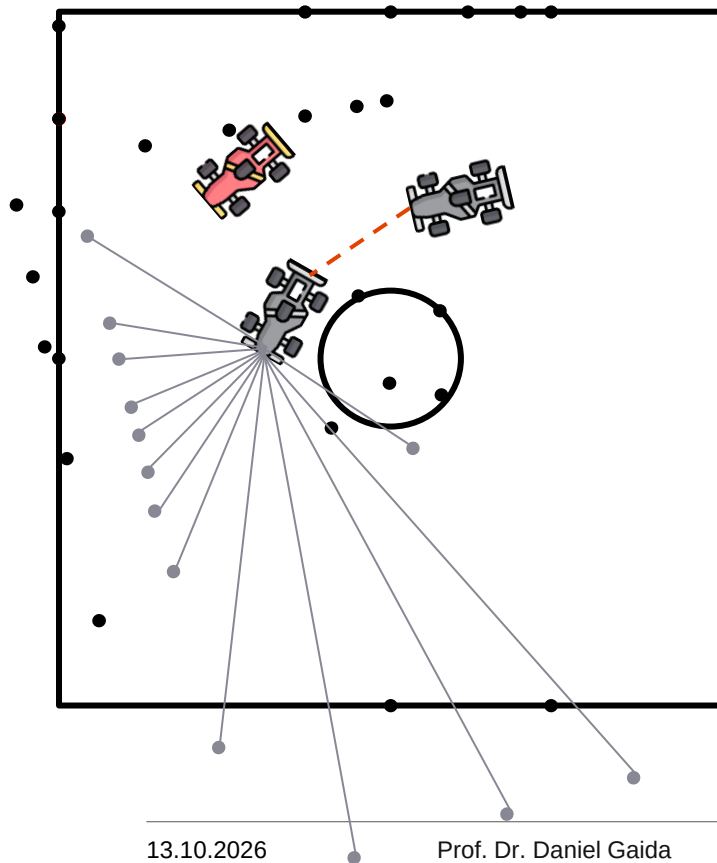
Pose Graphs



Pose Graphs



Pose Graphs



13.10.2026

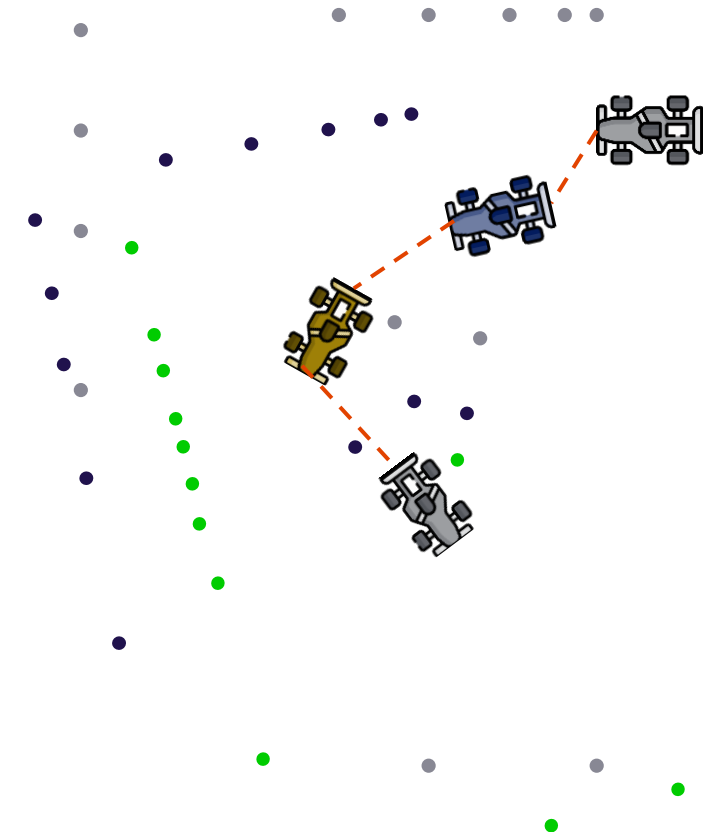
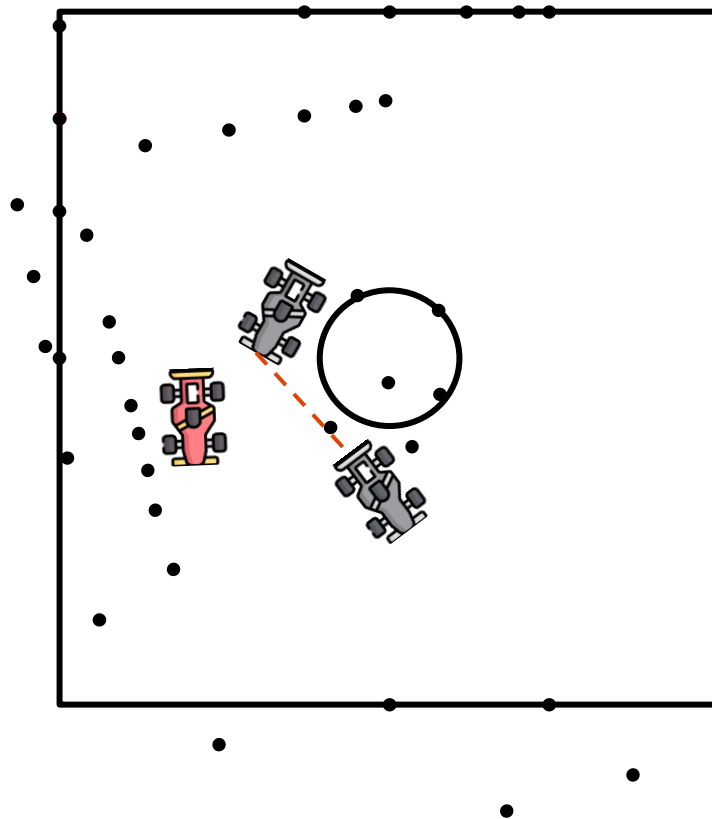
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Pose Graphs



13.10.2026

Prof. Dr. Daniel Gaida

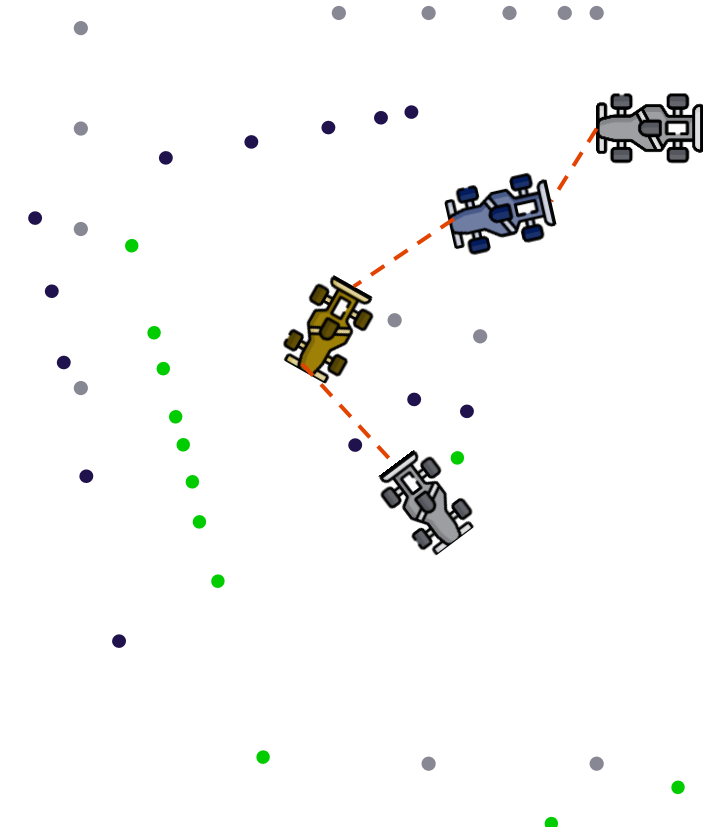
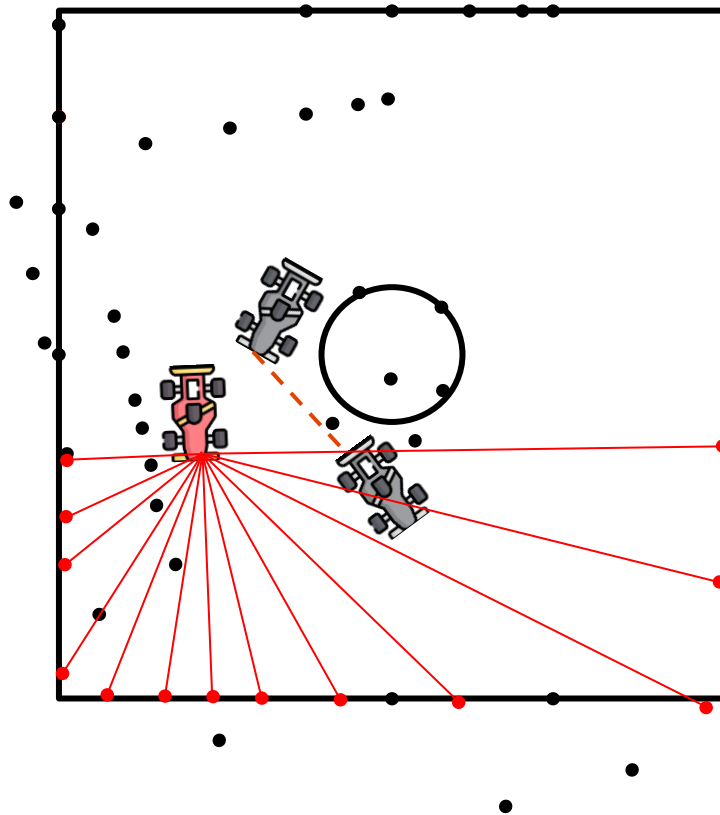
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

Pose Graphs



13.10.2026

• Prof. Dr. Daniel Gaida

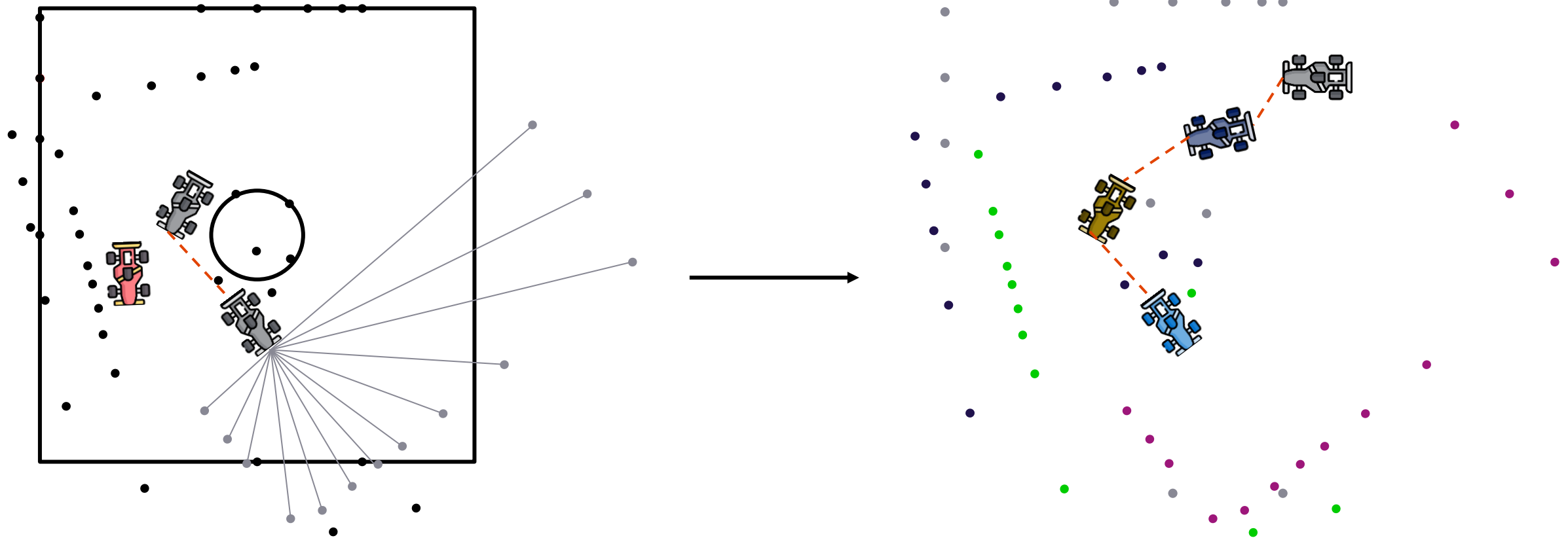
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

Seite 440

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Pose Graphs



13.10.2026

Prof. Dr. Daniel Gaida

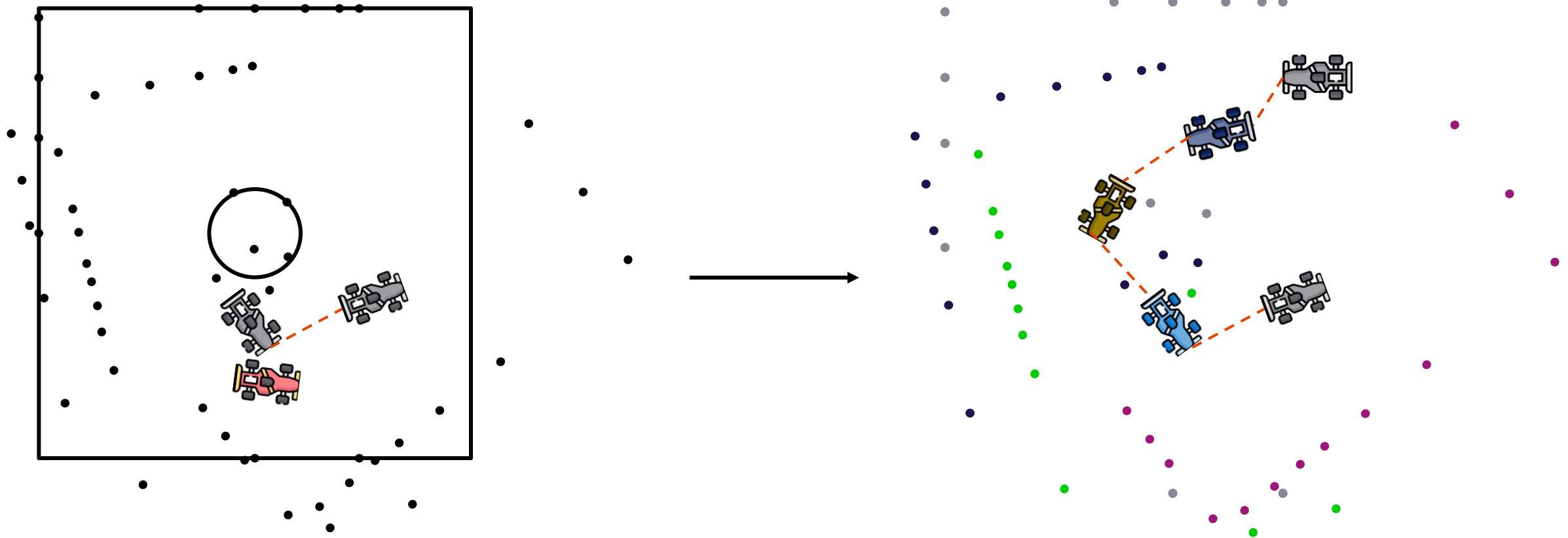
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

Pose Graphs



13.10.2026



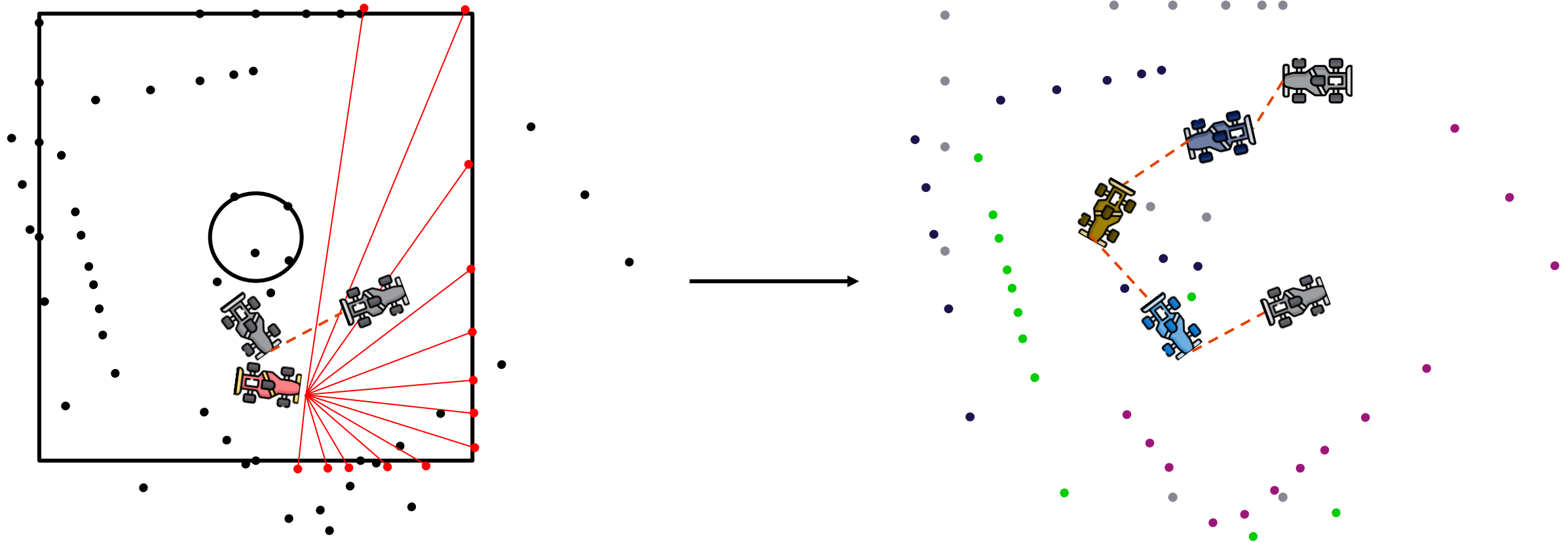
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Pose Graphs



13.10.2026

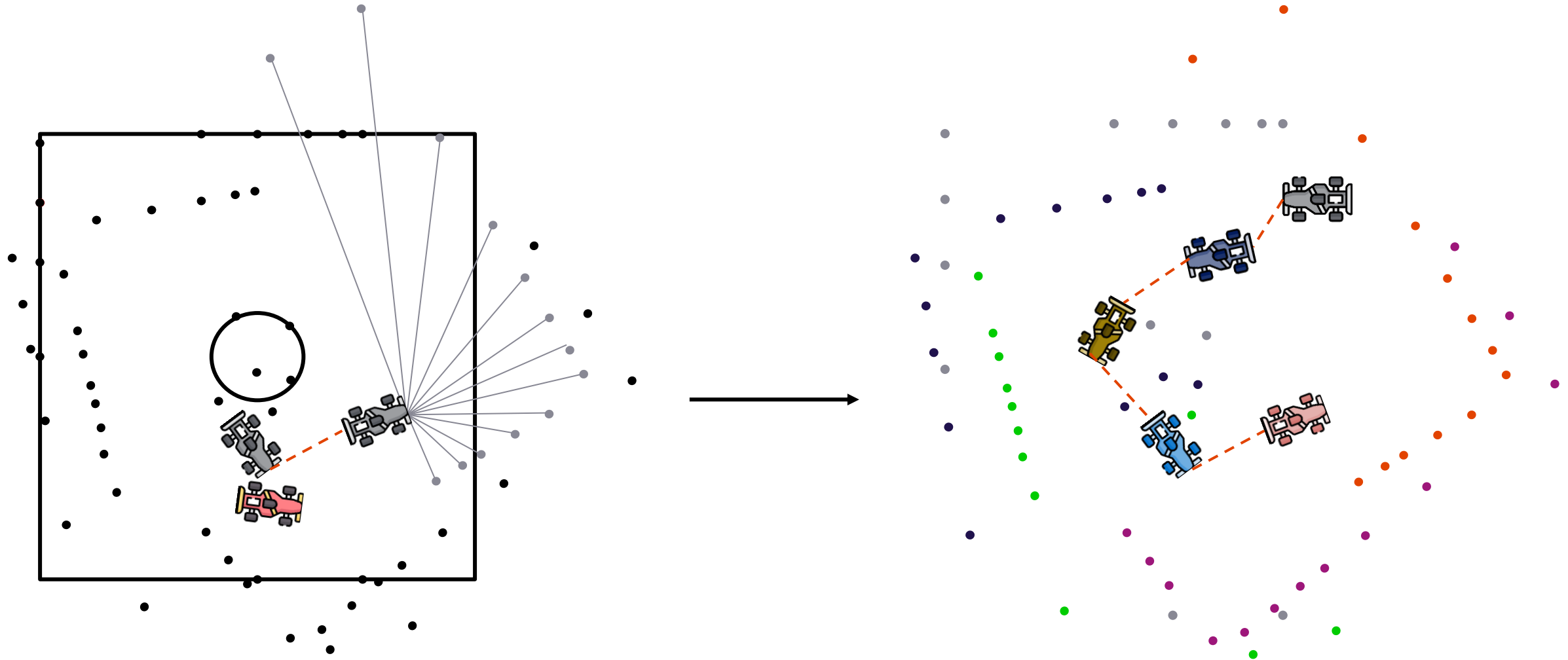
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Pose Graphs



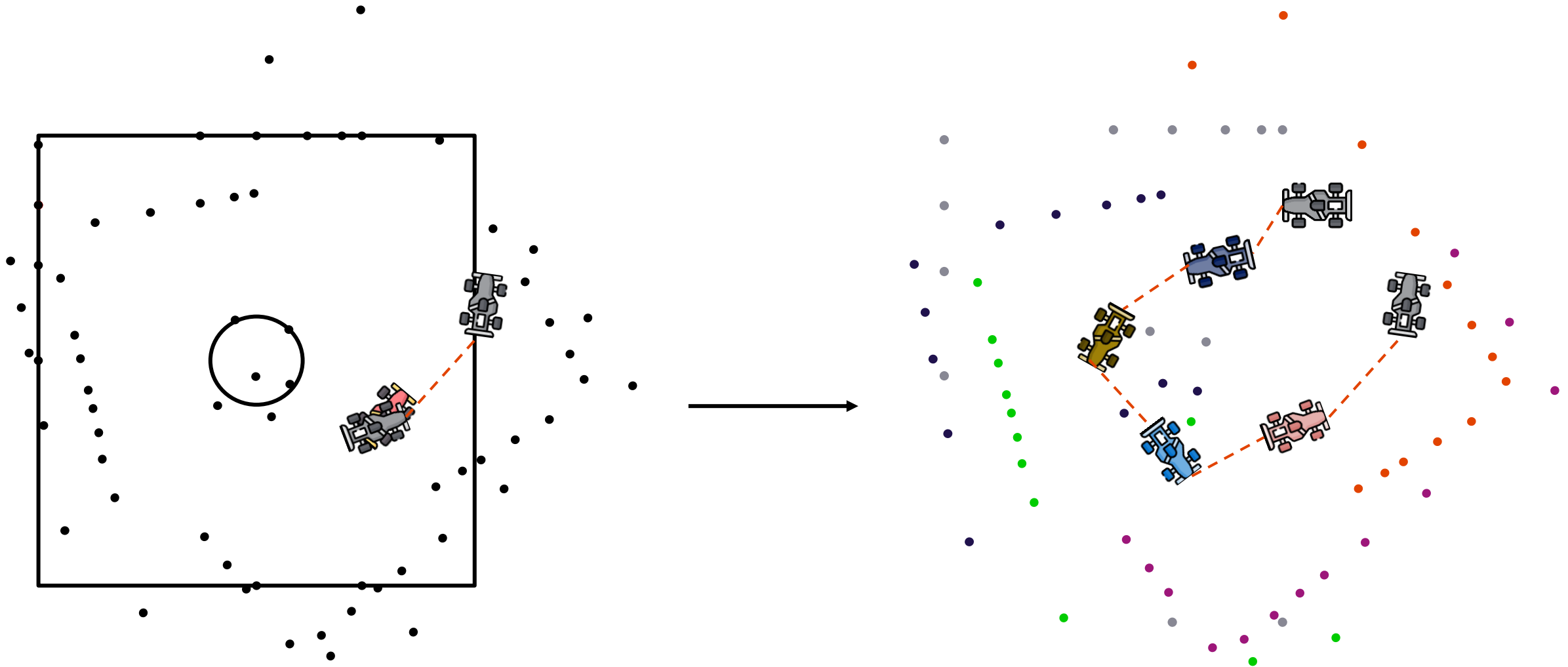
13.10.2026

• Prof. Dr. Daniel Gaida
Professor für Cyber-Physische Systeme
Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

Pose Graphs



13.10.2026

Prof. Dr. Daniel Gaida

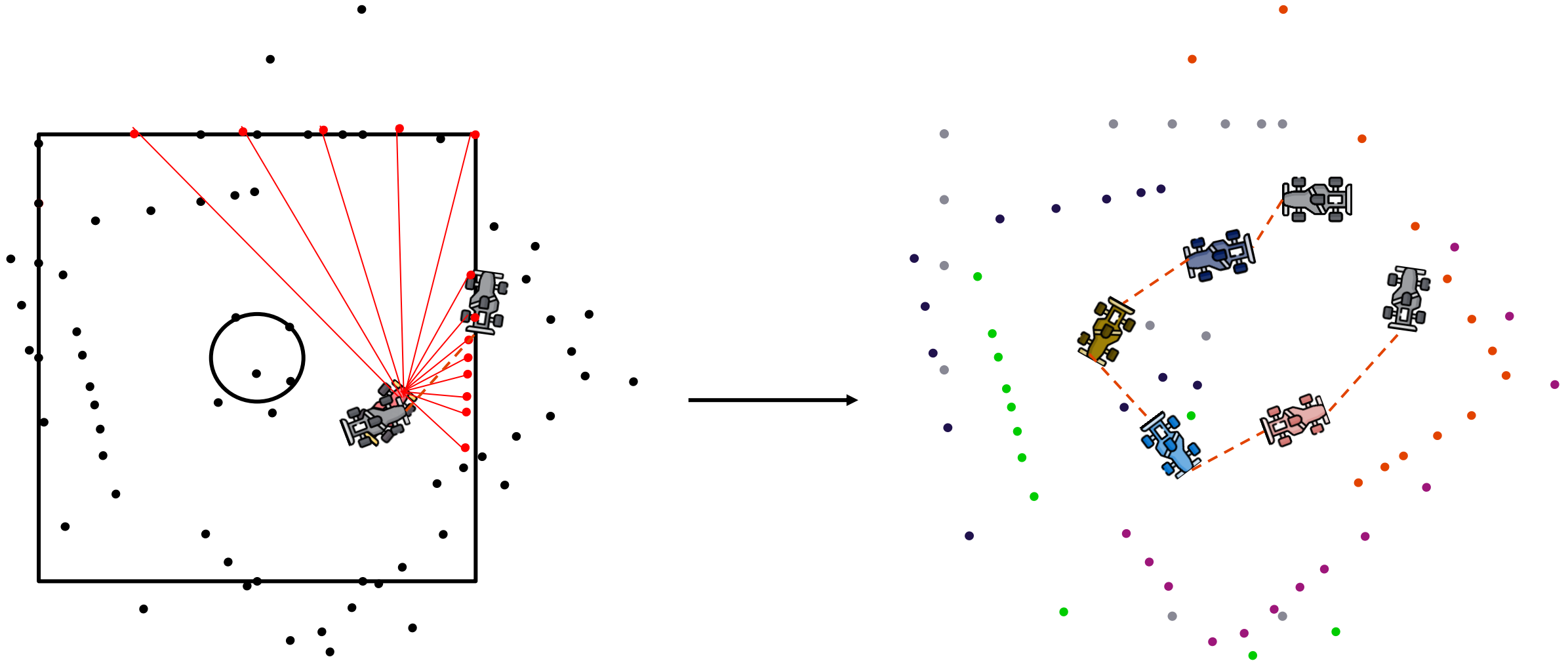
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

Pose Graphs



13.10.2026

• Prof. Dr. Daniel Gaida

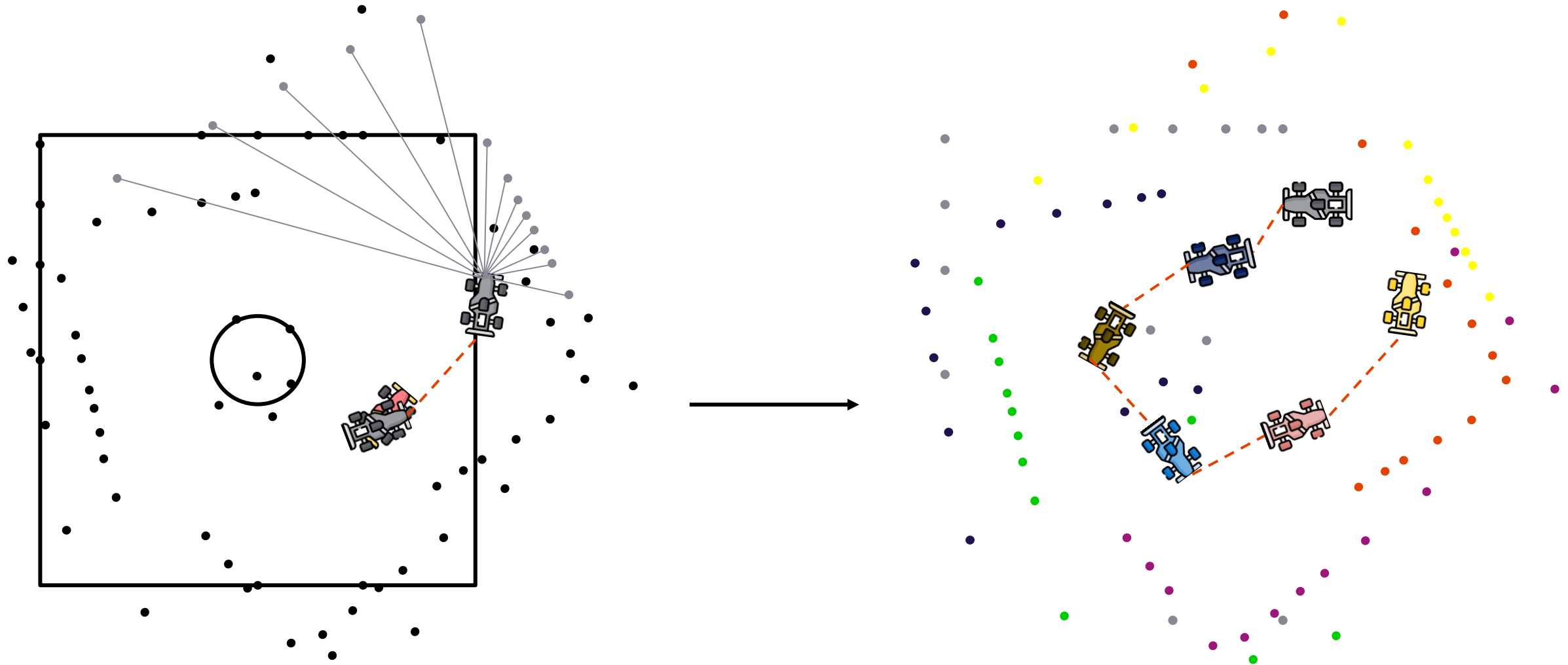
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

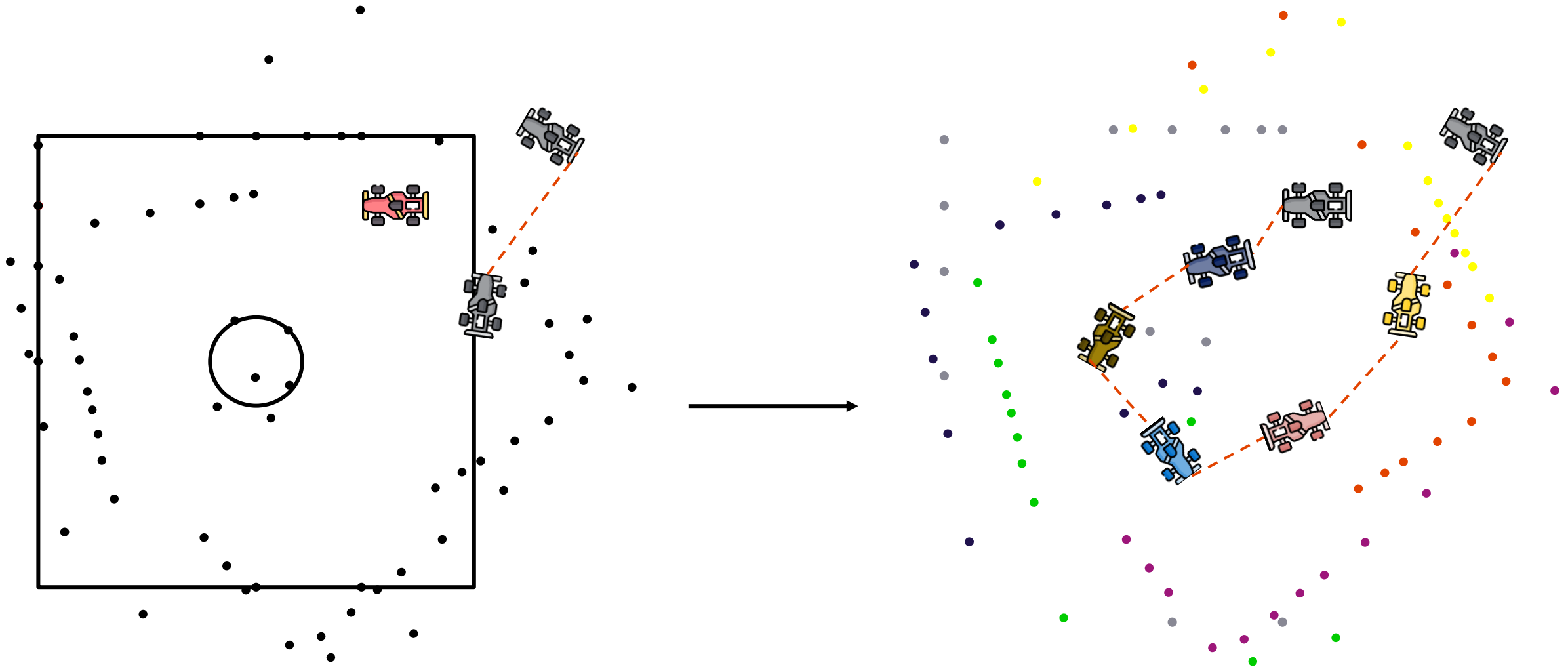
R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

Pose Graphs



Pose Graphs



13.10.2026

• Prof. Dr. Daniel Gaida

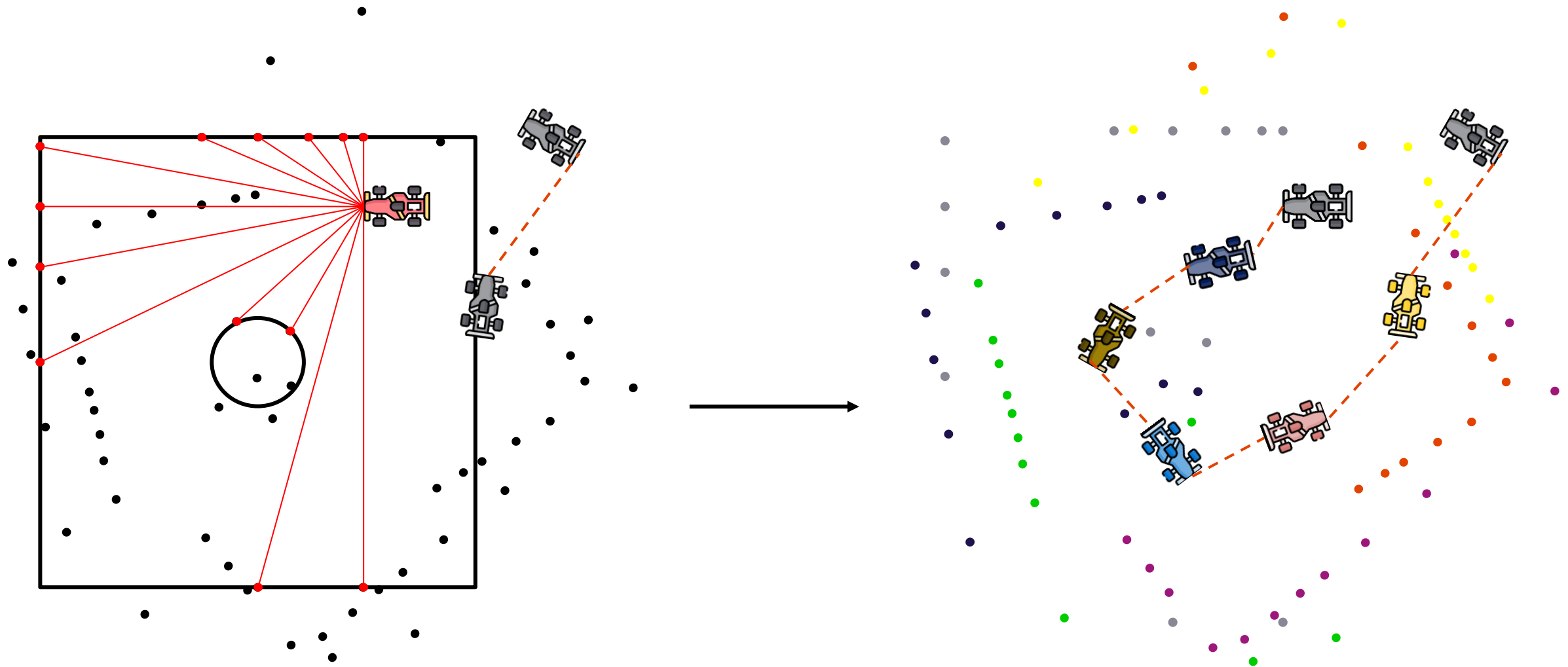
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

Pose Graphs



13.10.2026

Prof. Dr. Daniel Gaida

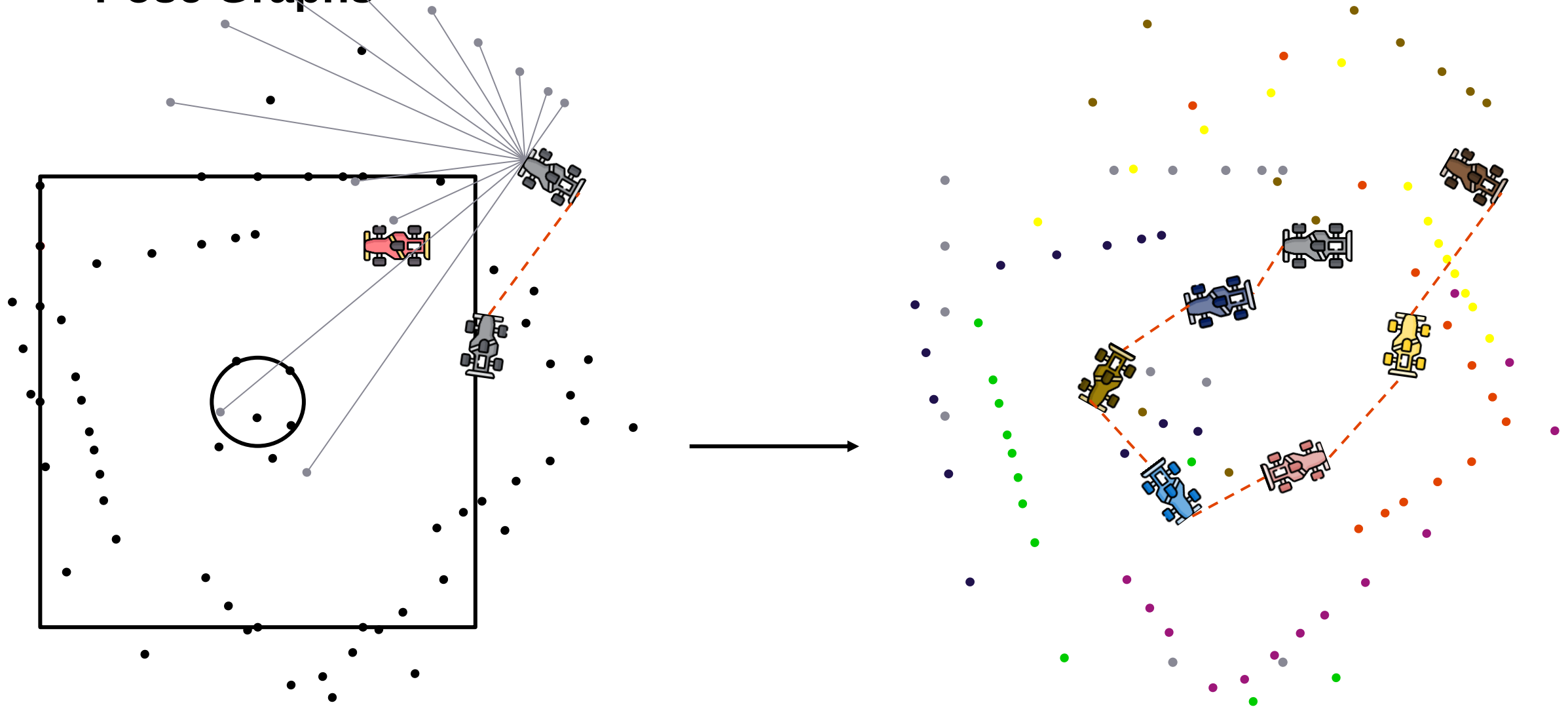
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

Pose Graphs



13.10.2026



Prof. Dr. Daniel Gaida

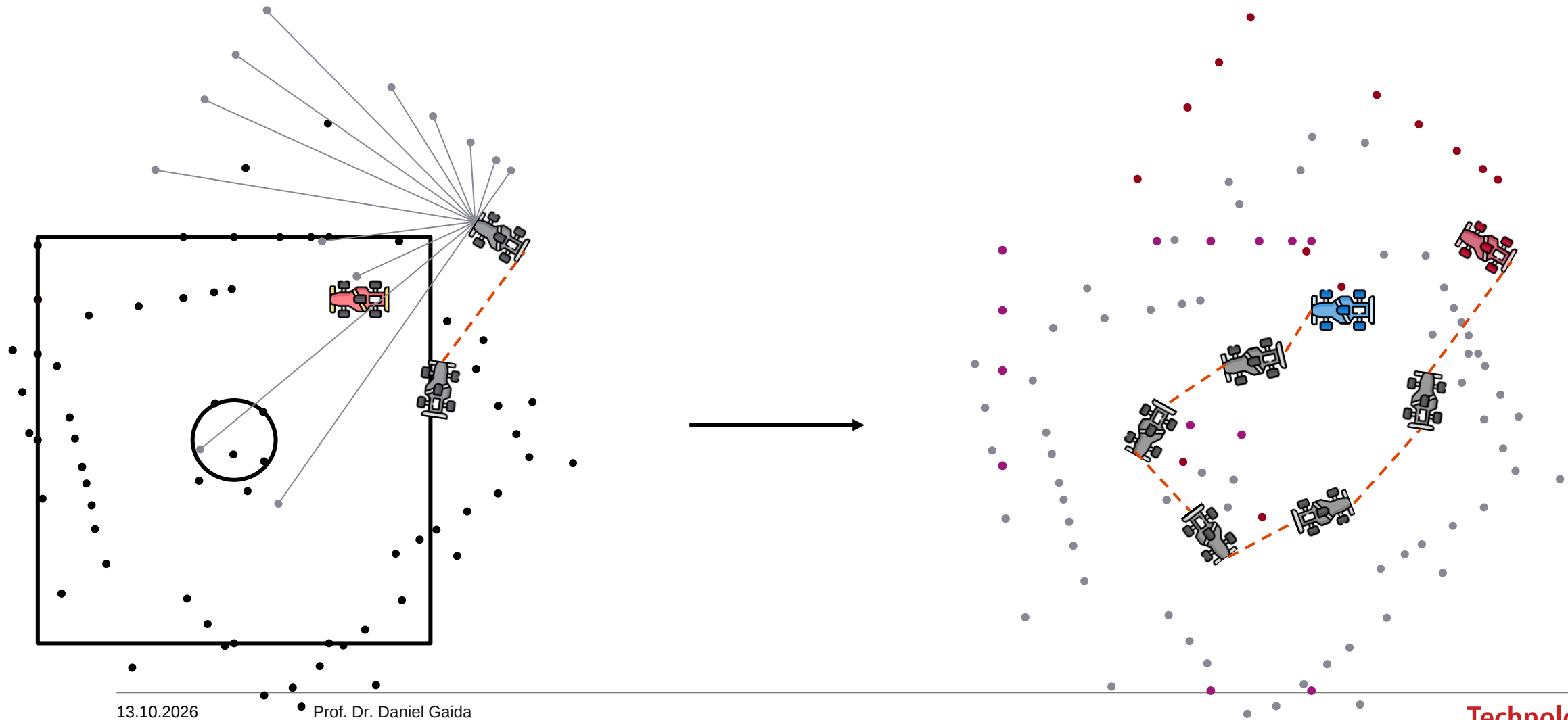
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

Seite 450

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Pose Graph Optimization: Loop closure



13.10.2026

Prof. Dr. Daniel Gaida

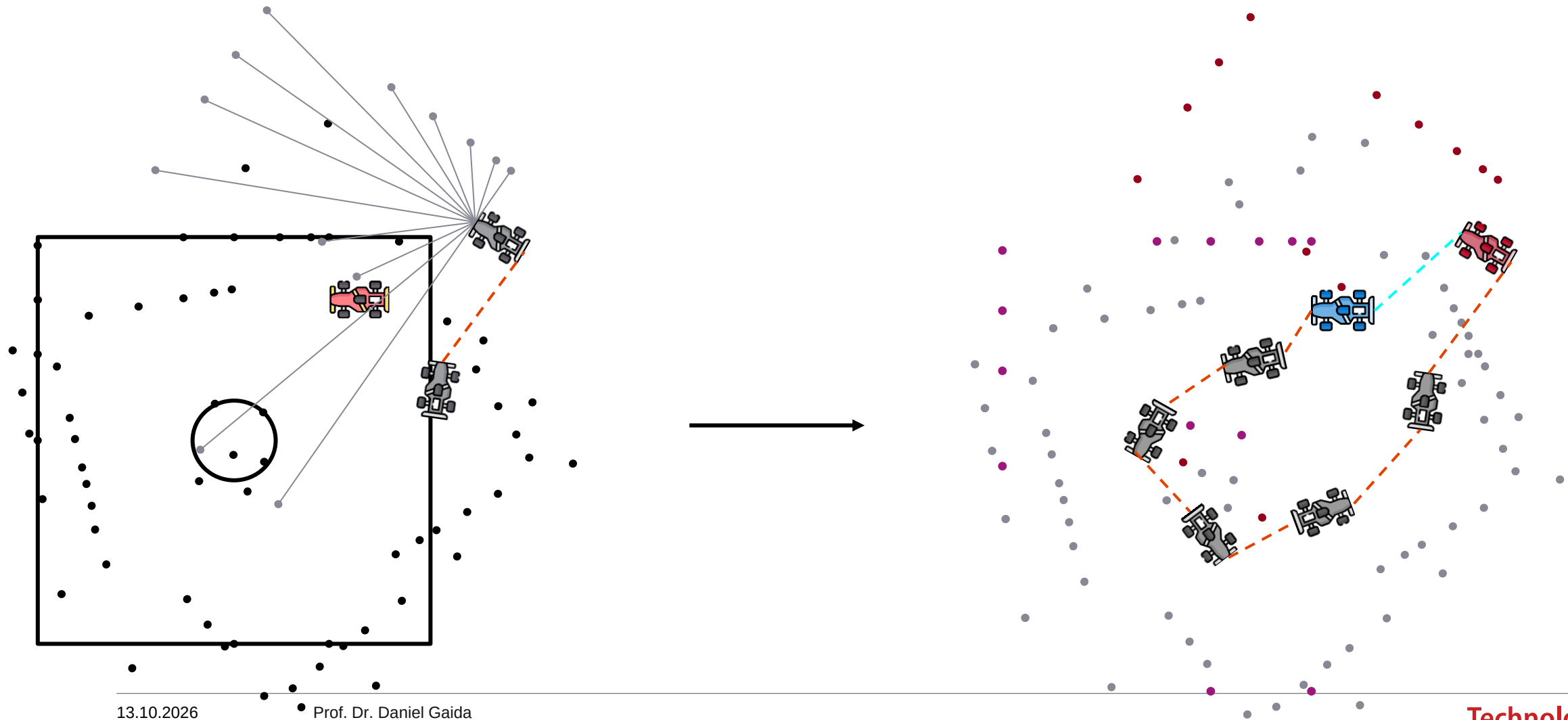
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

Pose Graph Optimization: Loop closure



13.10.2026

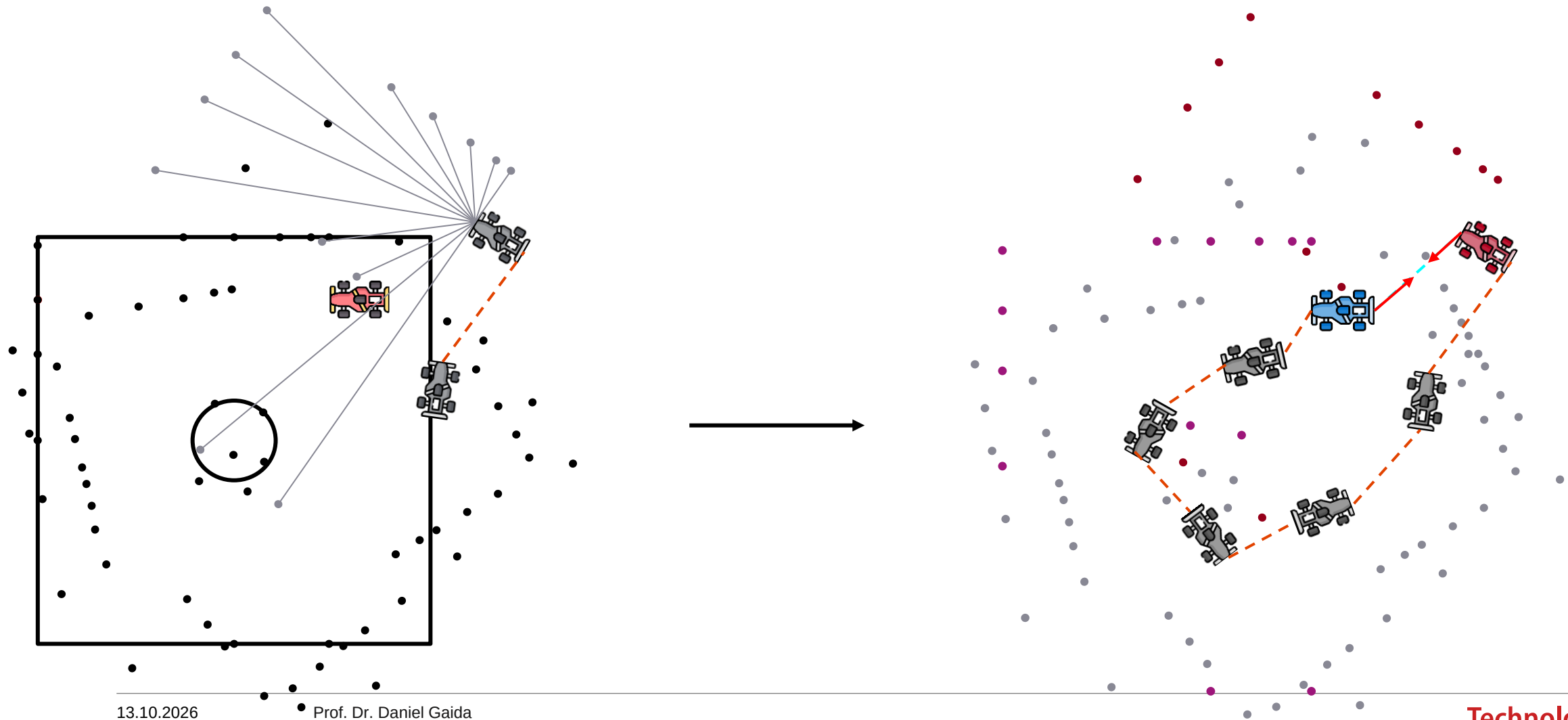
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Pose Graph Optimization: Loop closure



13.10.2026

Prof. Dr. Daniel Gaida

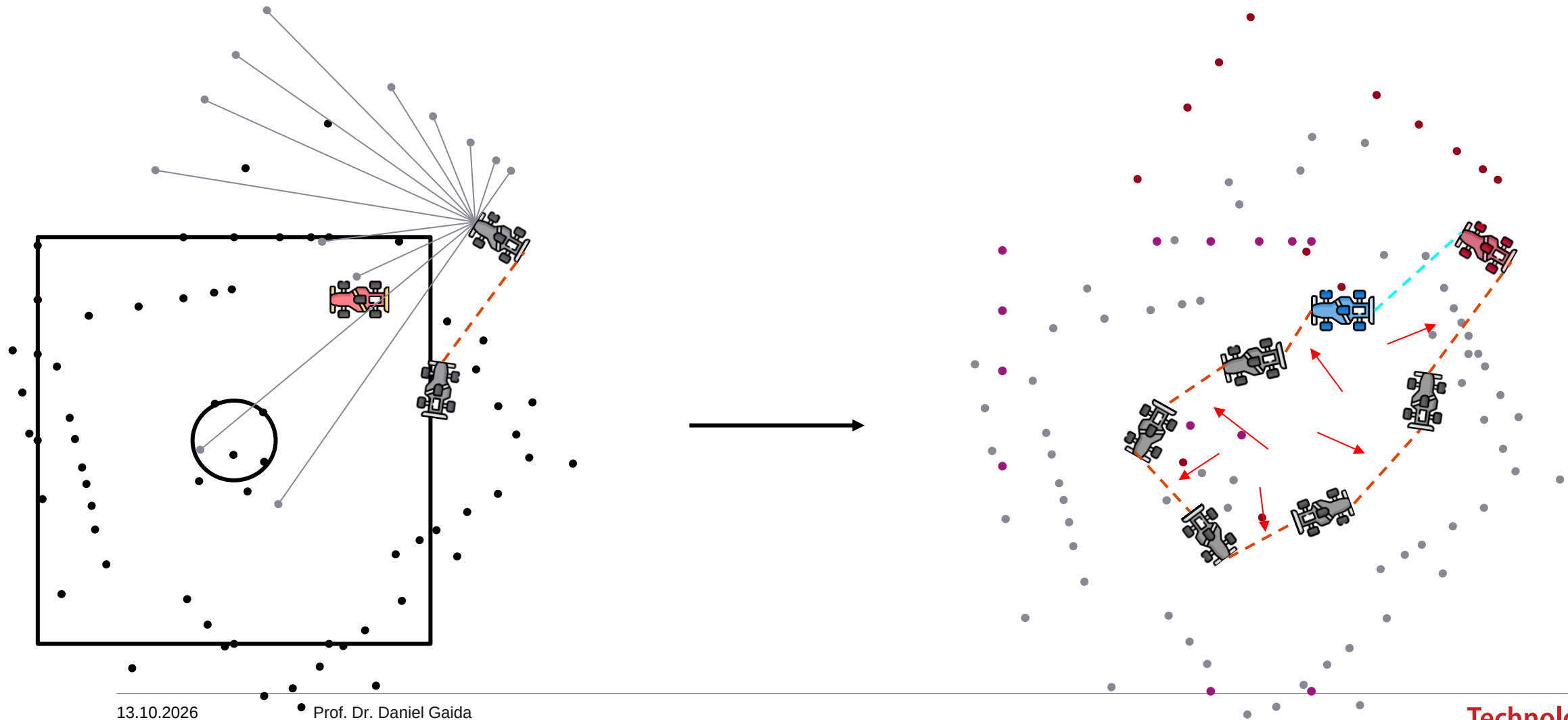
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

Pose Graph Optimization: Loop closure



13.10.2026

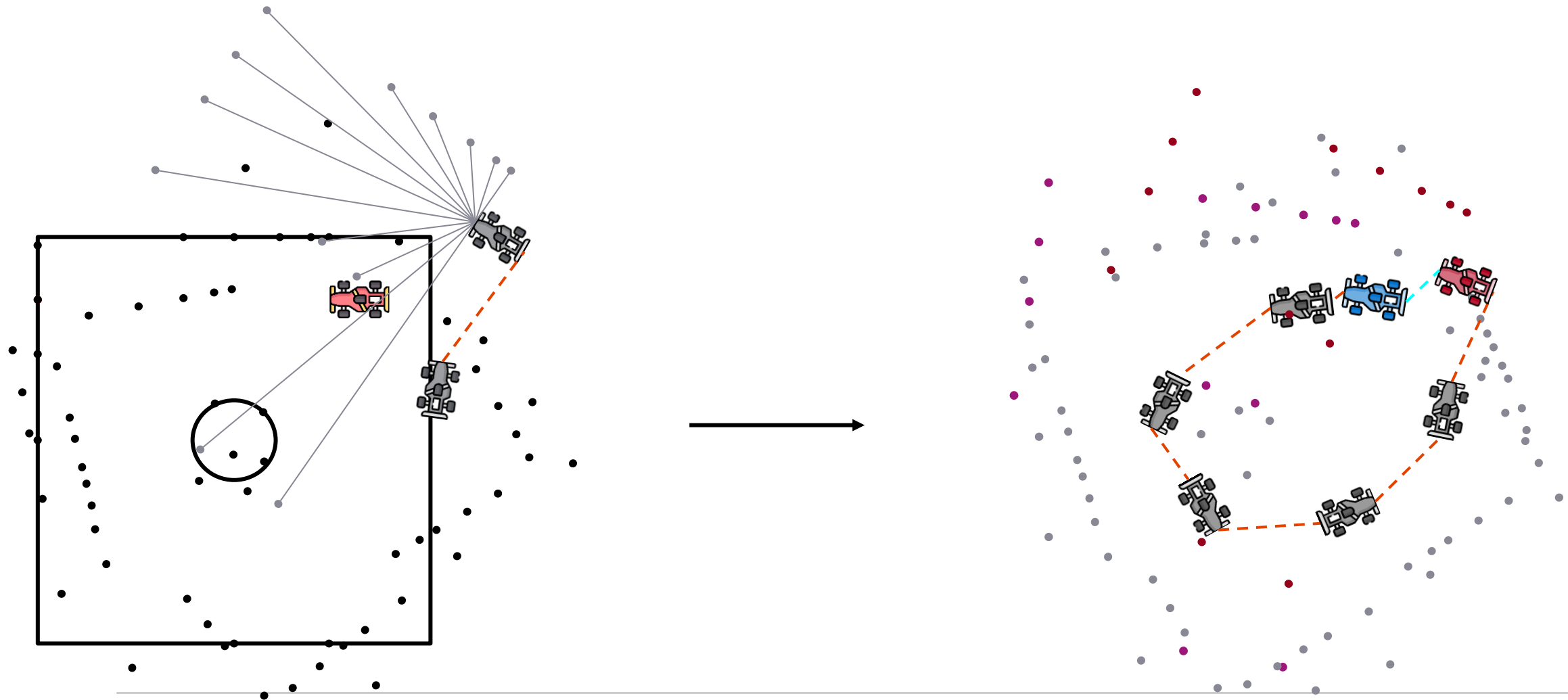
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Pose Graph Optimization: Loop closure



13.10.2026

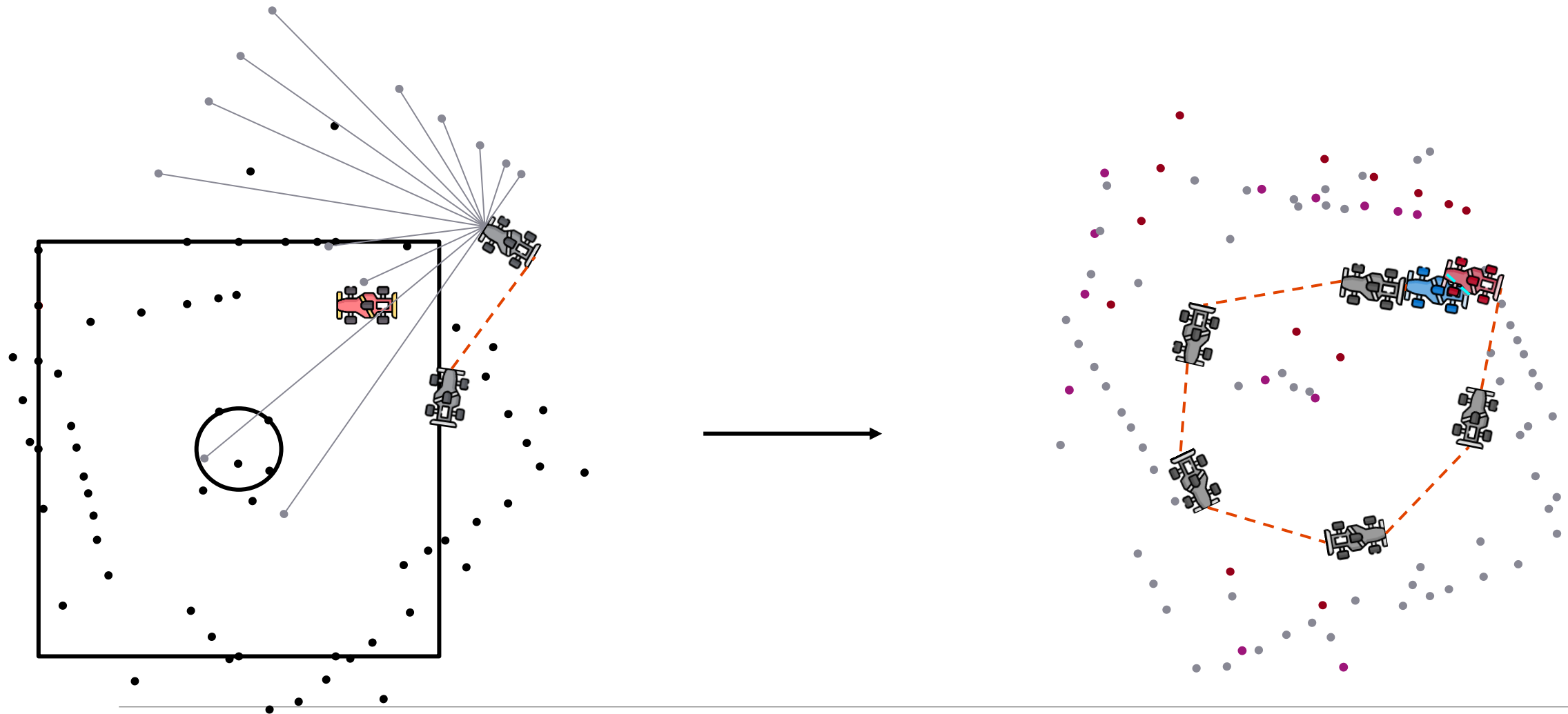
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Pose Graph Optimization: Loop closure



13.10.2026

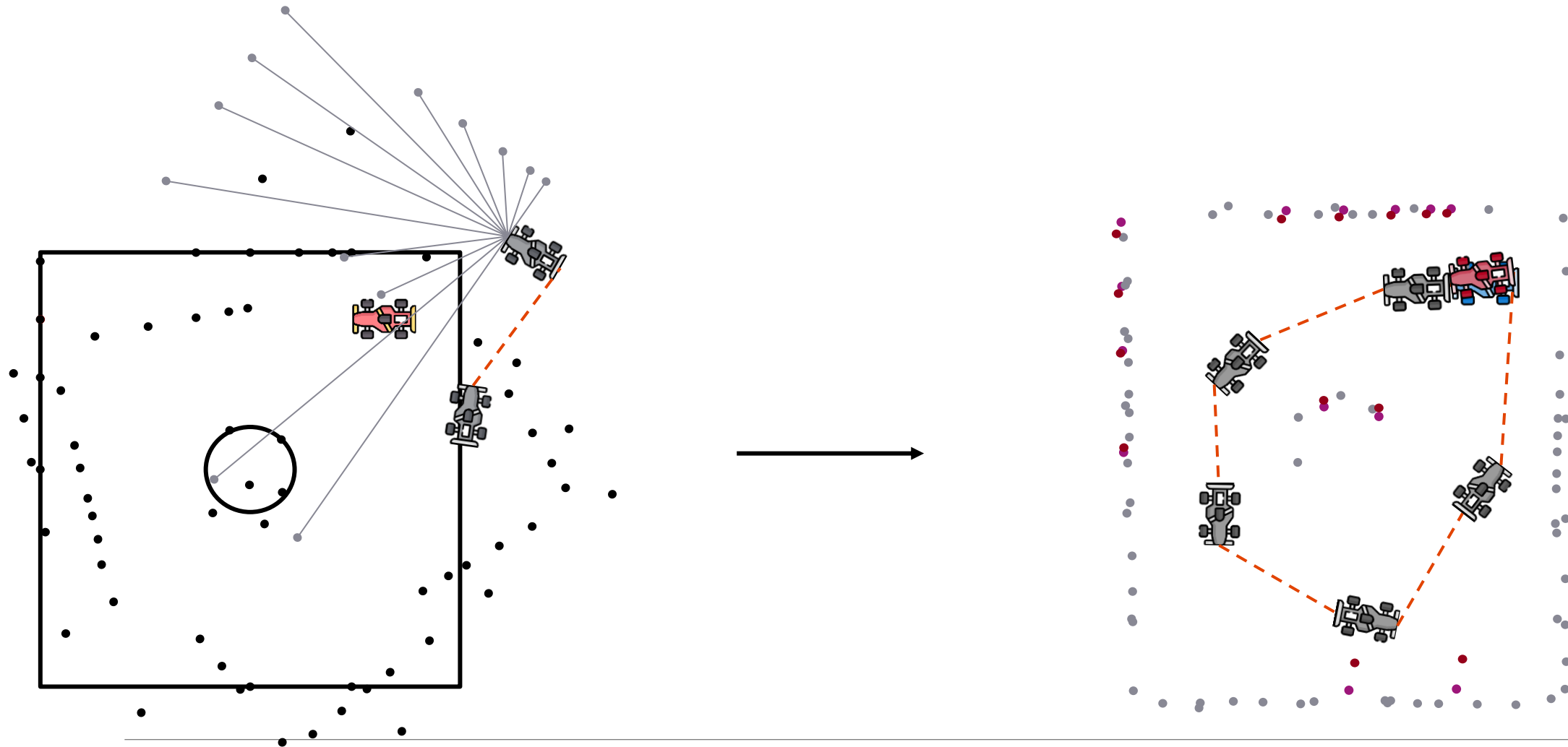
Prof. Dr. Daniel Gaida

Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Pose Graph Optimization: Loop closure



13.10.2026

Prof. Dr. Daniel Gaida

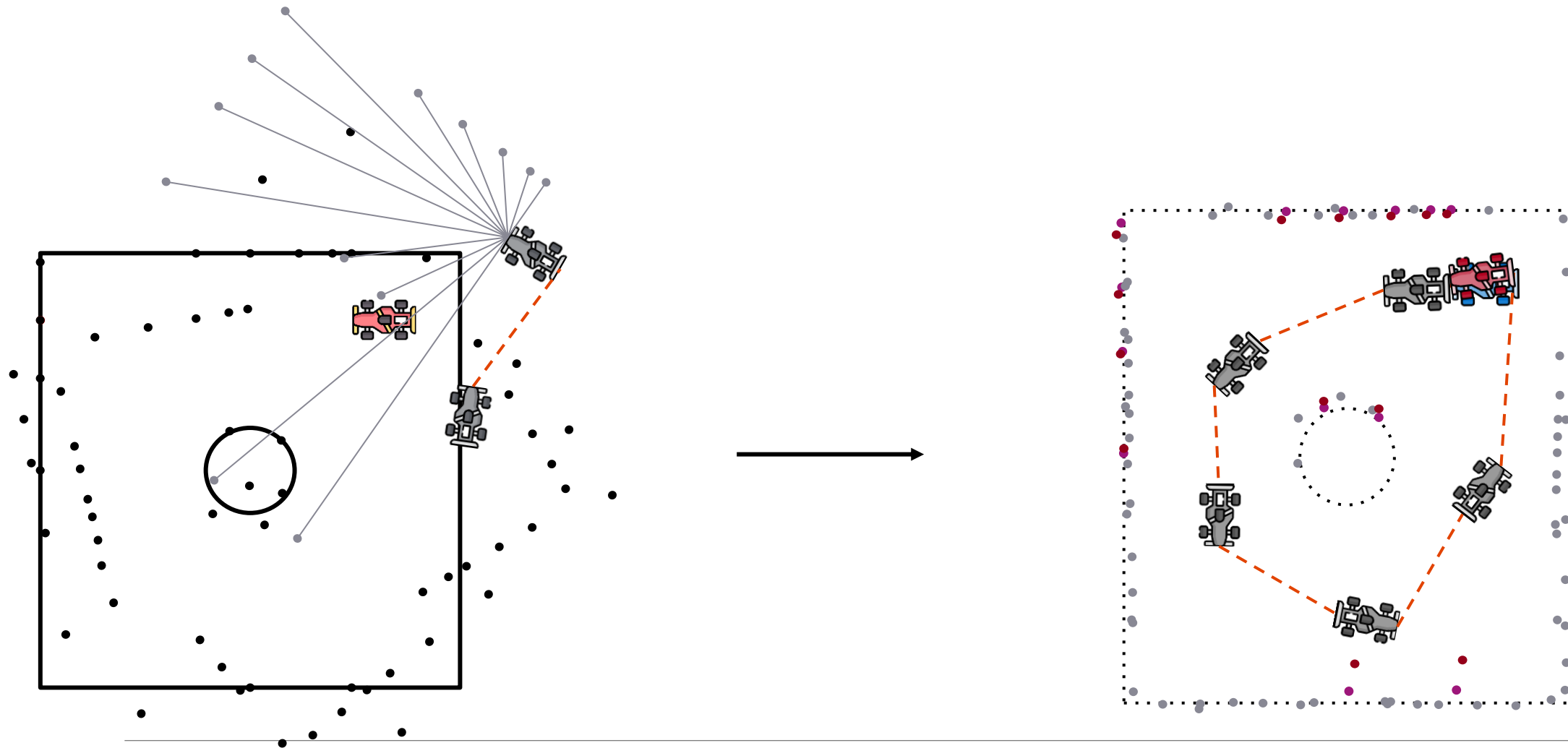
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

Technology
Arts Sciences
TH Köln

Pose Graph Optimization: Loop closure



13.10.2026

Prof. Dr. Daniel Gaida

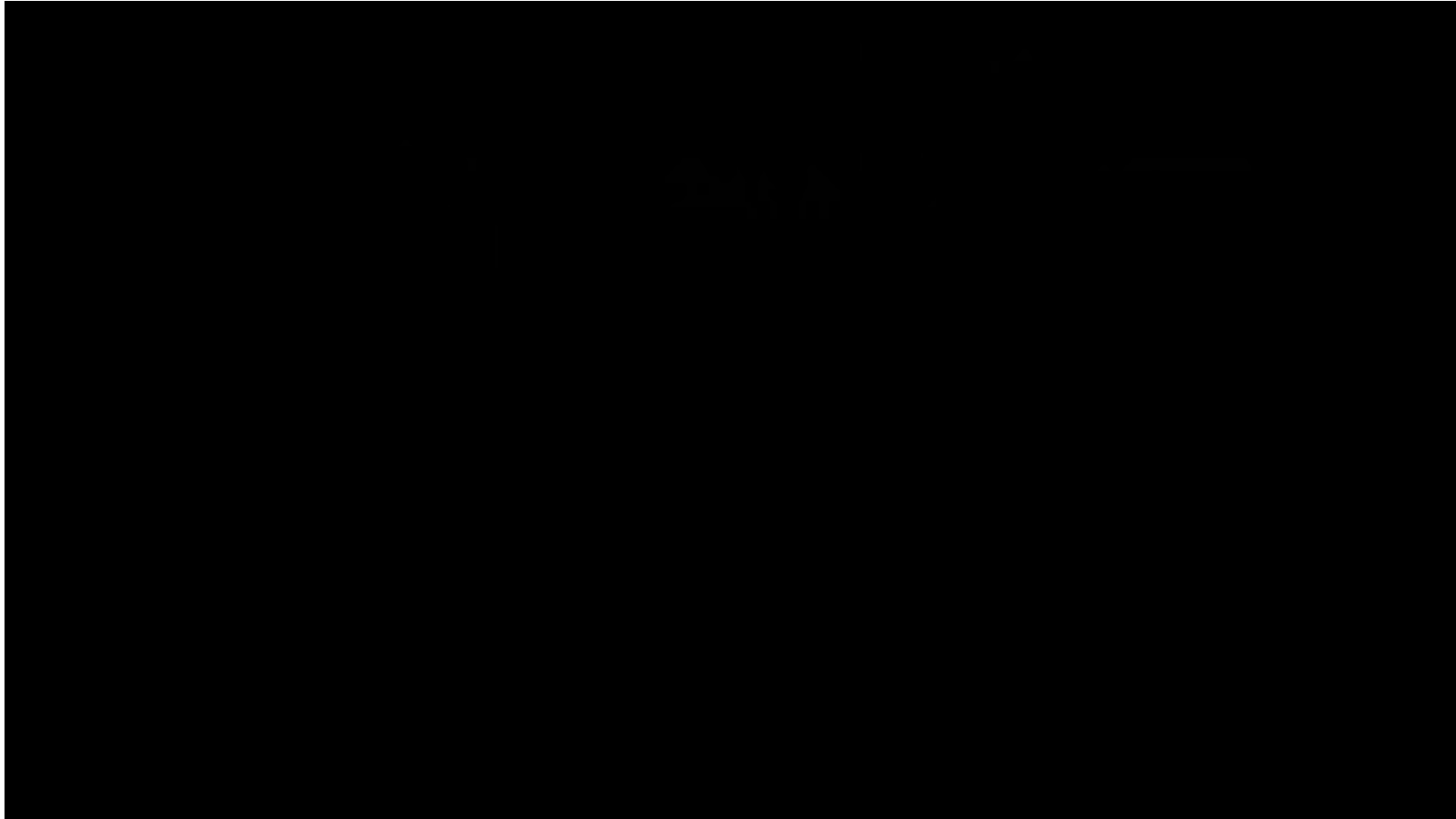
Professor für Cyber-Physische Systeme

Fakultät für Informatik und Ingenieurwissenschaften - Institut für Informatik

R. Mangharam, F1tenth Autonomous Racing, Penn Engineering,
University of Pennsylvania

SLAM - Simultaneous Localization and Mapping

SLAM - Simultaneous Localization and Mapping



SLAM - Zusammenfassung

Graph-basiertes SLAM ist sensorunabhängig. Es werden nur 2 Arten von Sensoren benötigt:

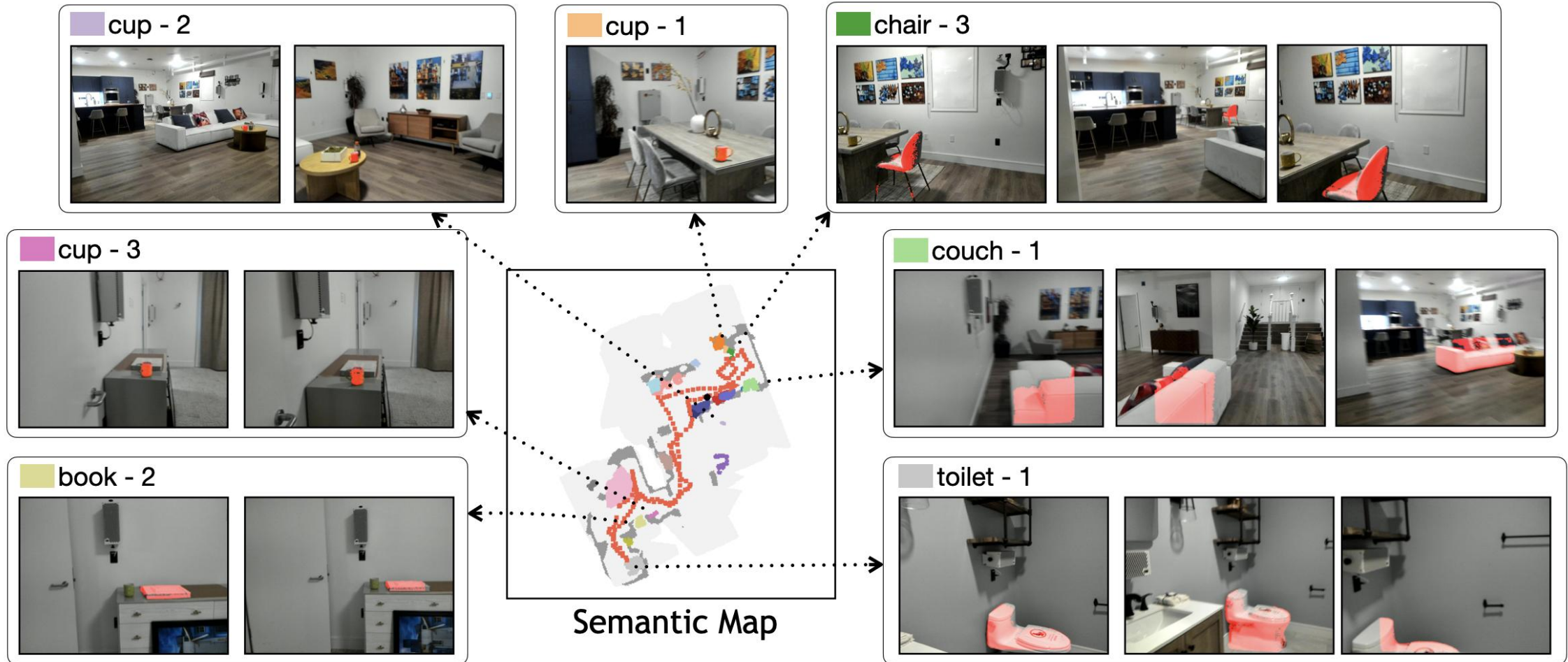
- Einen, der relative Transformationen zwischen Posen (Position + Orientierung) messen kann (IMS, Odometrie, Wheel-Encoder usw.). Dieser Sensor wird verwendet, um Beschränkungen zwischen den berechneten Posen zu erstellen.
- Ein Sensor, der die absolute Beobachtung der Umgebung messen kann. (LiDARs, Kameras, usw.). Dieser Sensor wird für das Schließen von Schleifen (Loop Closure) verwendet.

Inhalt

Modellierung der Umgebung von CPS:

- Modellierung mit KI/Deep Learning
 - Objektdetektion, -tracking, Bildsegmentierung, Greifpunkte
 - Warum ist die Modellierung der Umgebung mit nur einer Kamera schwierig?
 - Spracherkennung
 - Verhaltensmodellierung von Fußgängern, Fahrzeugen
 - World Models
- Lernen in virtuellen Umgebungen (in einer Simulation)
 - 3D Simulation
 - Von der Simulation in die Realität (Sim2real)
- **Statische Modelle der Umgebung (Karte)**
 - SLAM – Simultaneous Localization and Mapping
 - **Semantische Karte (Karte mit Objekten/Orten: Couch, TV, Küche, ...)**

Semantische Karte (Semantic Mapping)



Semantische Karte (Semantic Mapping)

Semantische Karte (Semantic Mapping)

Nächste Vorlesung und Fragen

- Robot Operating System
- Fragen?