Technical Workshop

Robust Perception Systems

Dr. Eng. Alexandru Forrai
Hassan Hotait

07-06-2023

SIEMENS

EU Guidelines on Robust & Trustworthy AI

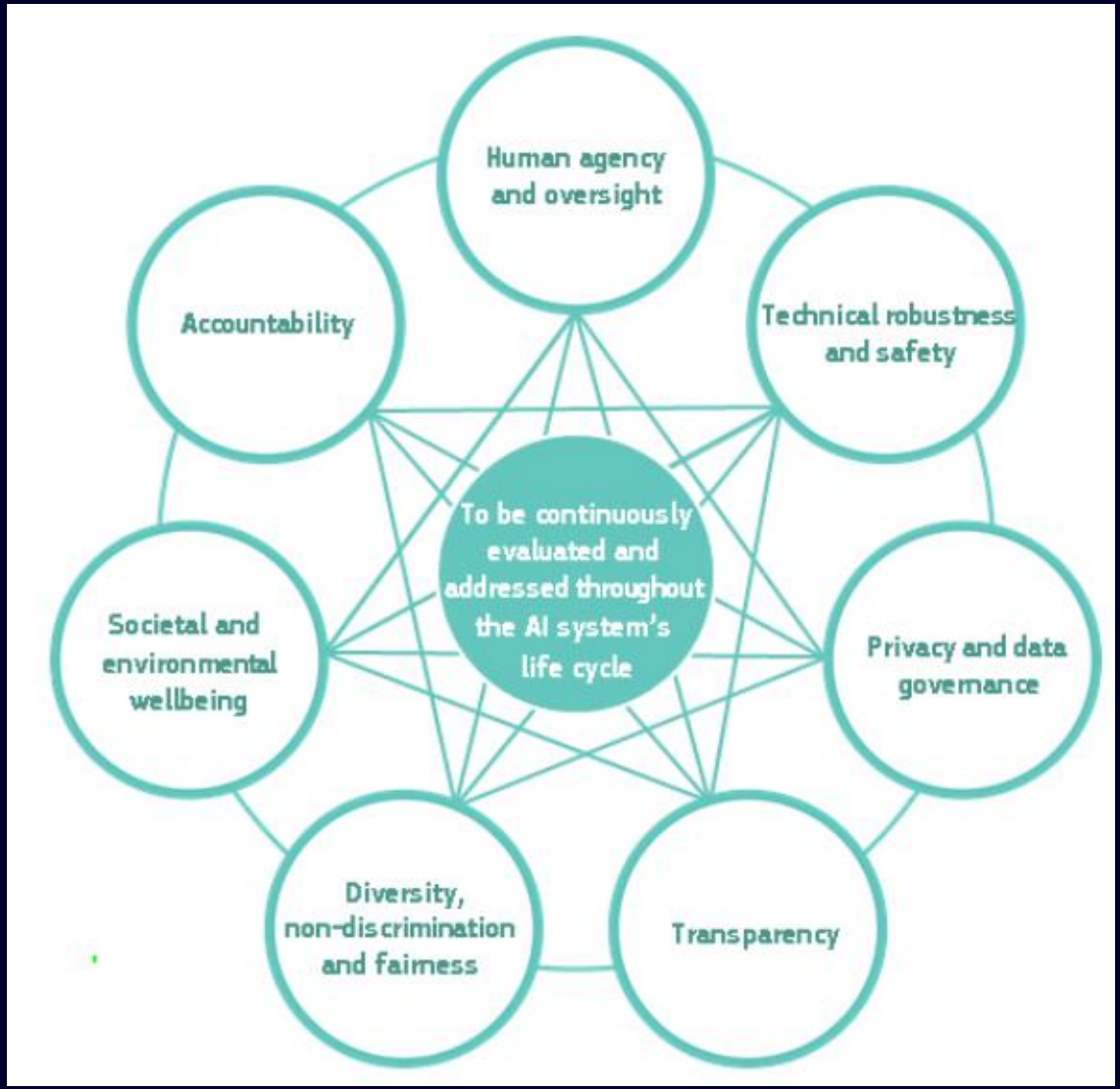


Figure 2: Seven Requirements to be implemented and evaluated In AI System Life cycle

SIEMENS

Contents

	Topic
1	Robust Perception Systems Introduction
2	Perception Stack: YOLOv3 & SMOKE
3	Evaluation Framework
4	Object Detection Evaluation Metrics
5	Robustness Assessment Framework
6	Localization & 3D Detection Assessment
7	SMOKE Localization/Depth Evaluation
8	Stereo Vision
9	Questions?

SIEMENS

2 Perception Stack: YOLOv3 & SMOKE

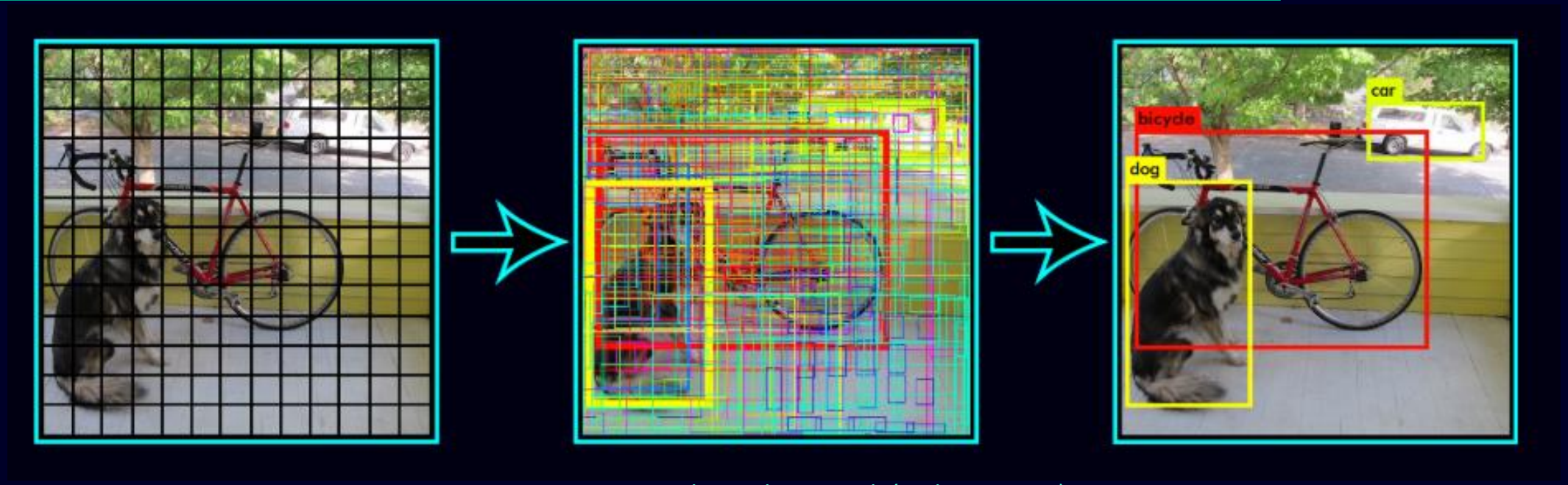


Figure 3: YOLOv3 Grid Based Approach (Redmon, 2018)

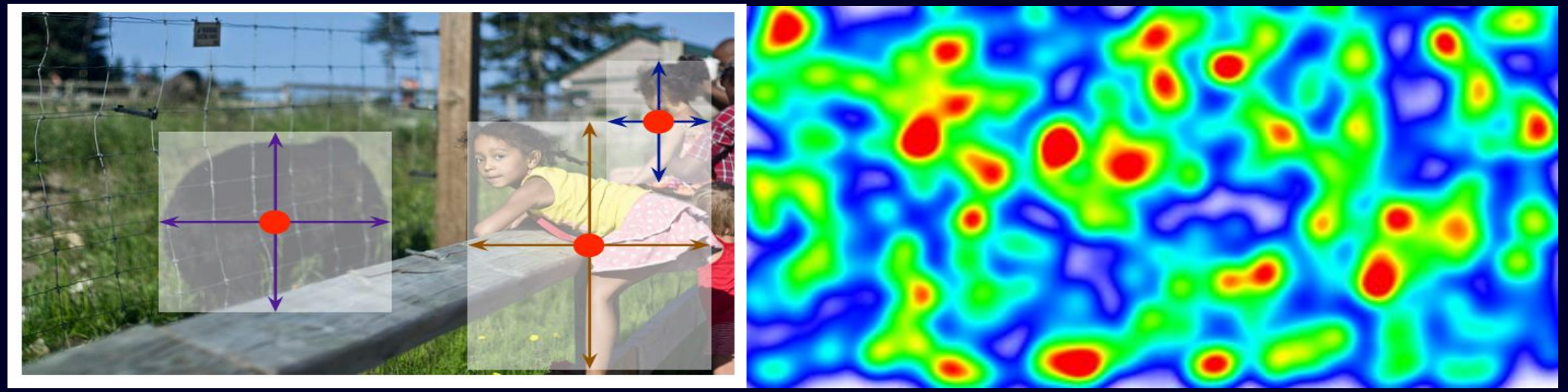


Figure 4: SMOKE Objects As Points Approach (Zechen Liu, 2019)

SIEMENS

2 Perception Stack: Demo

Figure 5: YOLOv3 Demo

Figure 6: SMOKE Demo

SIEMENS

3 Evaluation Framework

- **Labels & Predictions in KITTI Format are compared.**

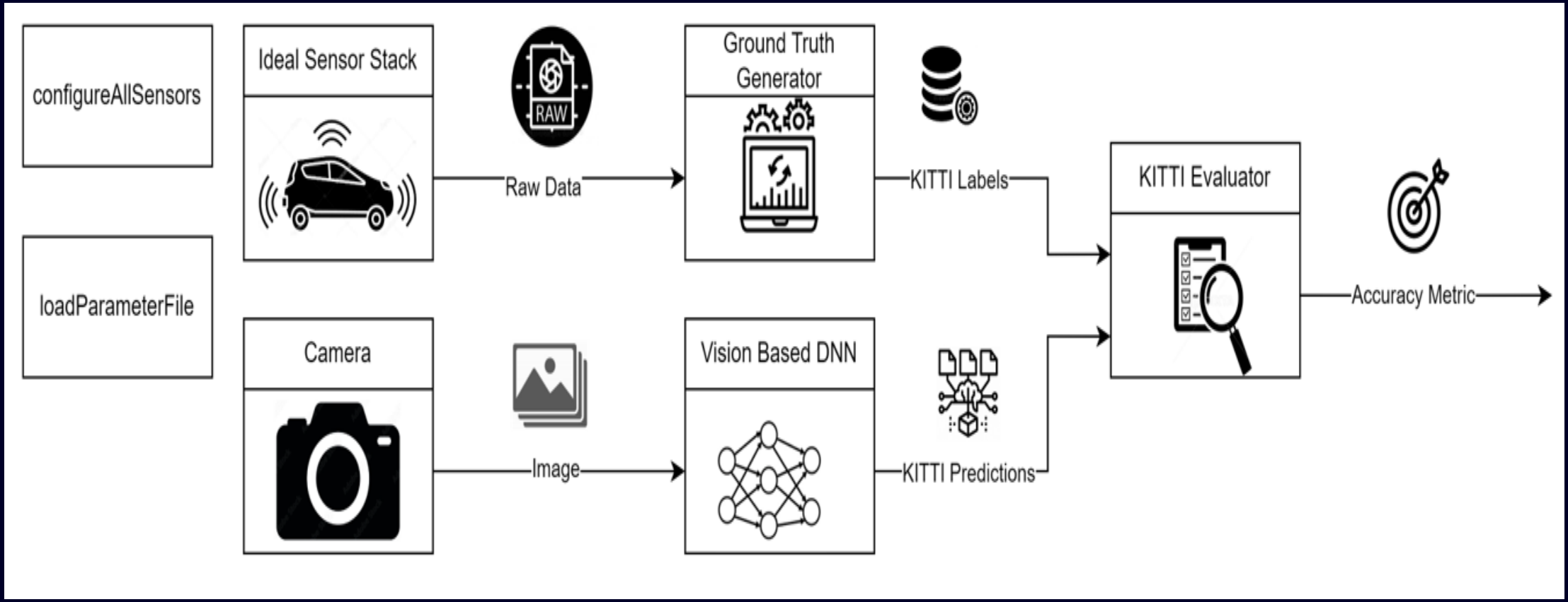


Figure 7: Evaluation Framework

SIEMENS

3 Evaluation Framework: KITTI Data Format

#Values	Name	Description
1	type	Describes the type of object: 'Car', 'Van', 'Truck', 'Pedestrian', 'Person_sitting', 'Cyclist', 'Tram', 'Misc' or 'DontCare'
1	truncated	Float from 0 (non-truncated) to 1 (truncated), where truncated refers to the object leaving image boundaries
1	occluded	Integer (0,1,2,3) indicating occlusion state: 0 = fully visible, 1 = partly occluded 2 = largely occluded, 3 = unknown
1	alpha	Observation angle of object, ranging [-pi..pi]
4	bbox	2D bounding box of object in the image (0-based index): contains left, top, right, bottom pixel coordinates
3	dimensions	3D object dimensions: height, width, length (in meters)
3	location	3D object location x,y,z in camera coordinates (in meters)
1	rotation_y	Rotation ry around Y-axis in camera coordinates [-pi..pi]
1	score	Only for results: Float, indicating confidence in detection, needed for p/r curves, higher is better.

Figure 8: KITTI Data Format Showing Fields & Values

3 Evaluation Framework: Ground Truth Generation Pipeline

Prescan ground truth sensors allow us to obtain all the information needed for labels in KITTI data format.

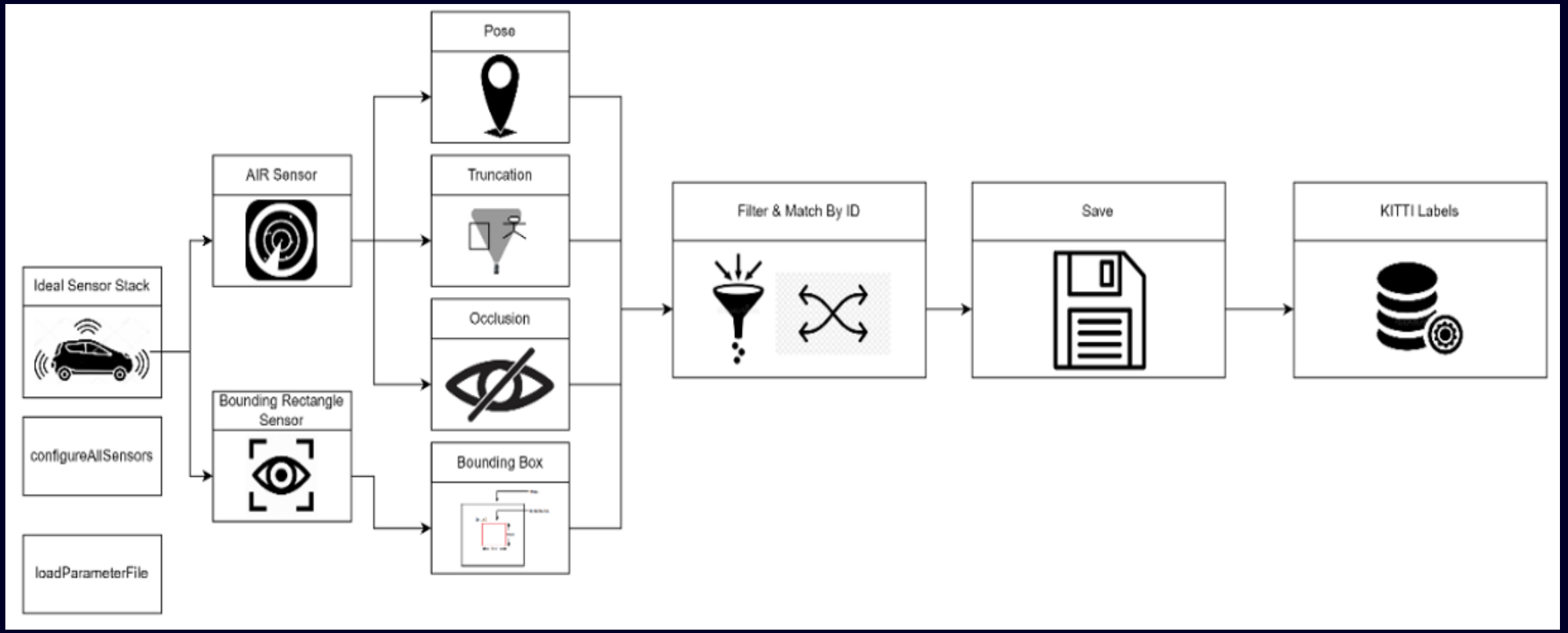


Figure 9: Ground Truth Generation Pipeline

SIEMENS

Evaluation Framework: Ground Truth Generator AIR Sensor Limitation

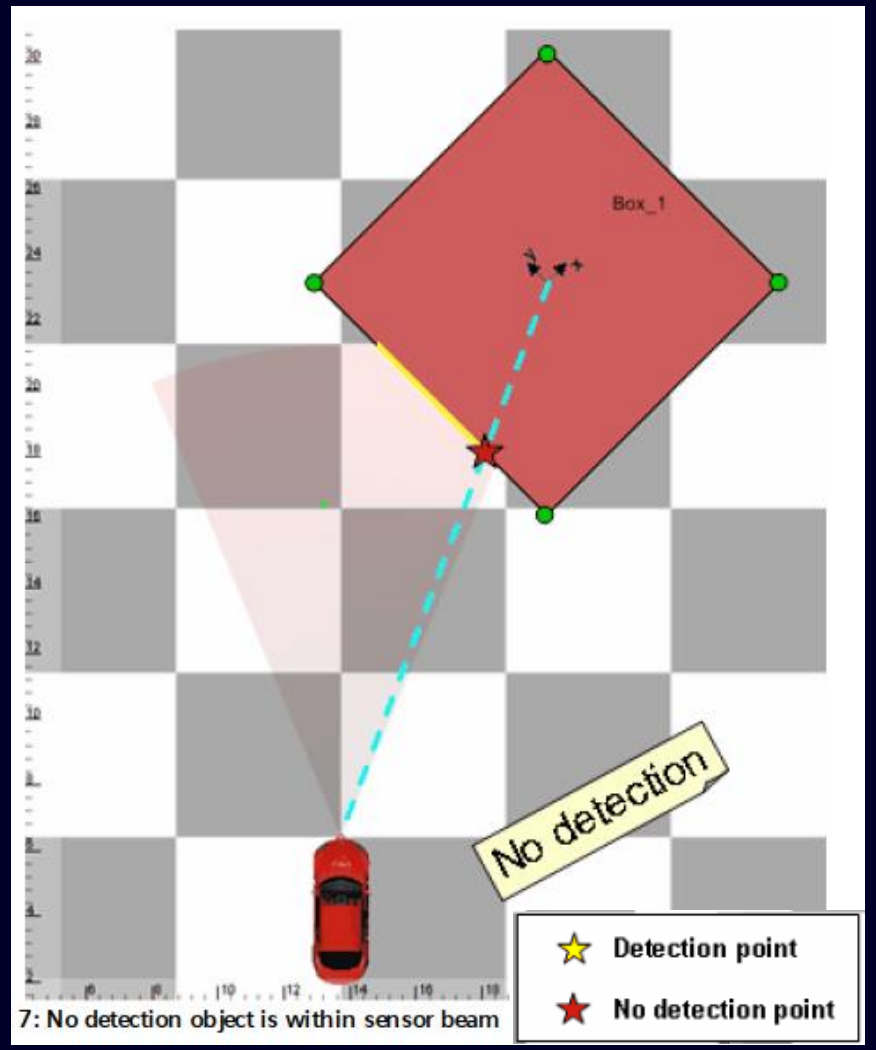


Figure 10: Effect of Detection Type on AIR Sensor behavior.

AIR Detection Type:

- *Center: Measures Distance to Center of Object*

3 Evaluation Framework: Ground Truth Generator Truncation Limitation

Algorithm 1: Truncation

Input: X, Y, lengthObject,widthObject,rotGlobal, sensorFOV,sensorRange
Output: truncation
Procedure: *Get intersection point of rectangle representing object with sensor field of view. Use rectangle corner points to get the distances u and v. Truncation is the ratio of object within FOV over complete object.*

Get Rectangle Corner Points
N = Get Number of Corner Points within FOV

if $N = 0$ **then**
  $truncation \leftarrow 1$;
end
else if $N = 4$ **then**
  $truncation \leftarrow 0$;
end
else
  Get coordinates of diagnoal with vertex outside FOV.

  Find intersection point of FOV with diagonal.

  Given the coordinates, compute u and v.

  $truncation \leftarrow \frac{u}{u+v}$;
end

Figure 11: Algorithm for computing truncation.

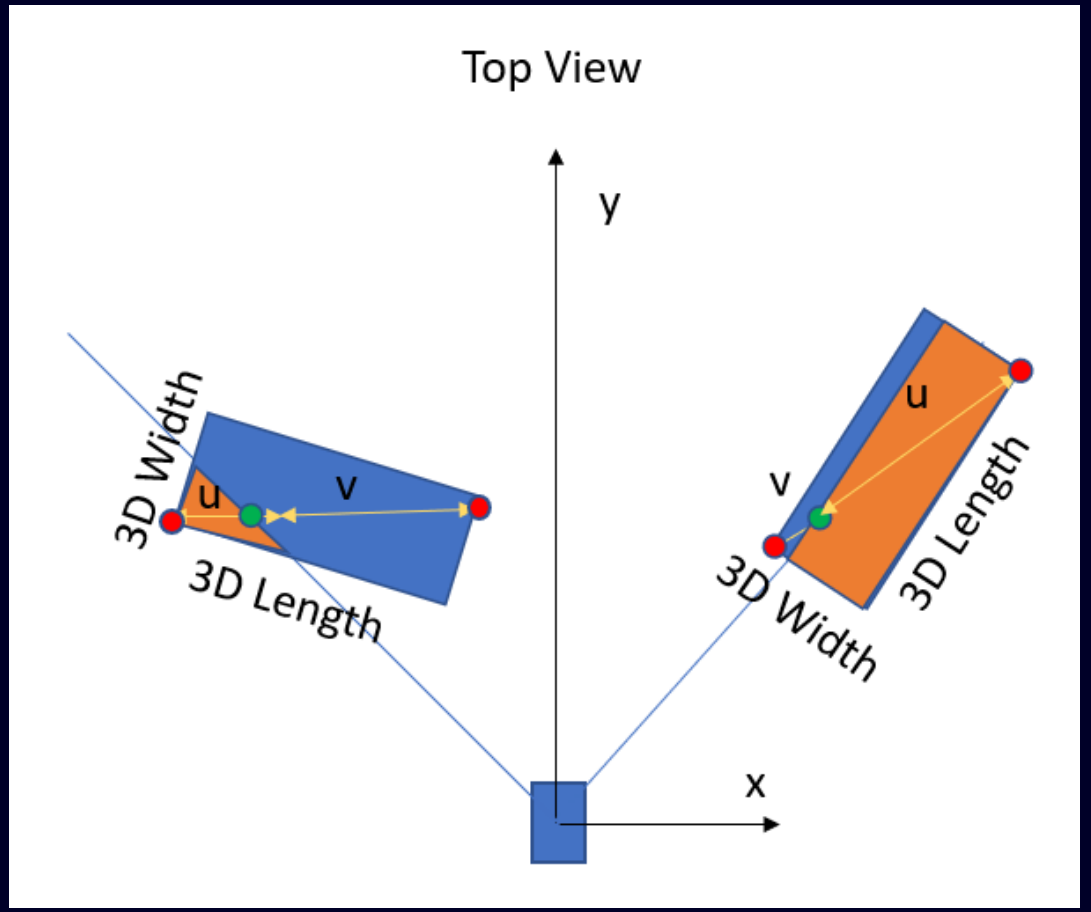


Figure 12: 2D Truncation Representation

SIEMENS

3 Evaluation Framework: Ground Truth Generator Occlusion Limitation

Algorithm 1 Occlusion

Input: X, Y, lengthObject,widthObject,rotGlobal, sensorFOV,sensorRange
Output: occlusion
Procedure: *Get the number of rays that hit the object with and without occlusion. The ratio and the thresholds are used to assign to occlusion level*

N = Number Of Rays
n = 0
m = 0

for *i = 1 to N* **do**
  // Perform operations within the loop instructions
  if *Ray ∩ Object @ No Occlusion* **then**
    $m + = 1$
  end
  if *Ray ∩ Object @ Occlusion* **then**
    $n + = 1$
  end
end

$occlusionRatio = \frac{n}{m}$
if *occlusionRatio* < 0.05 **then**
  $occlusion \leftarrow 0$
  ▷ Fully Visible
end
else if $0.05 <$ *occlusionRatio* < 0.5 **then**
  $occlusion \leftarrow 1$
  ▷ Partly Occluded
end
else if $0.5 <$ *occlusionRatio* < 0.75 **then**
  $occlusion \leftarrow 2$
  ▷ Largely Occluded
end
else
  $occlusion \leftarrow 3$
  ▷ Unkown
end

Figure 13: Algorithm for computing occlusion.

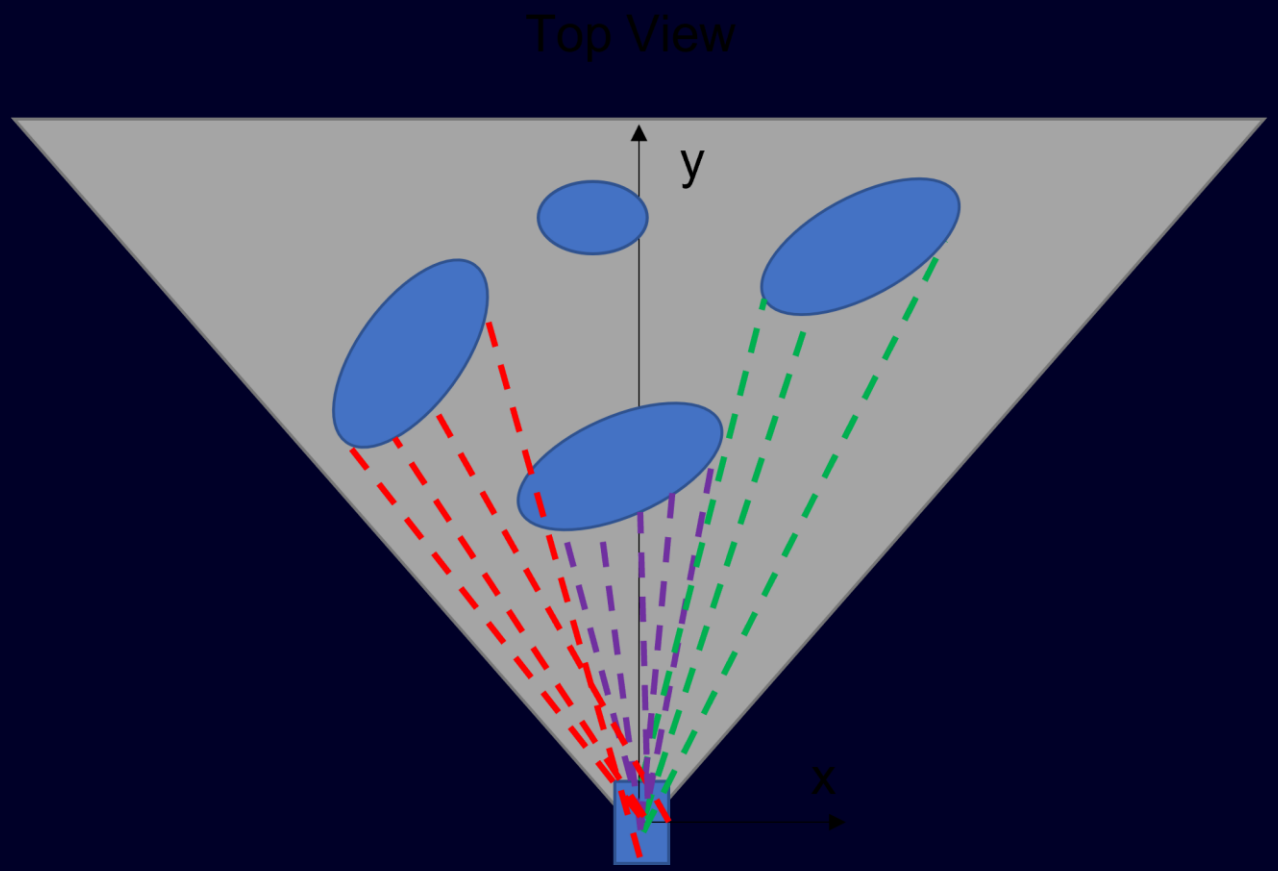


Figure 14: Rays from camera center hitting objects.

SIEMENS

4 Object Detection Evaluation Metrics

$$Accuracy\ (Ac) = \frac{TP+TN}{TP+TN+FN+FP}$$

TN = 0 → Object Detection

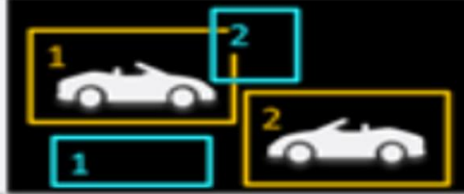
		Precision (Pr) $\frac{TP}{TP+FP}$	Recall (Rc) $\frac{TP}{TP+FN}$	Accuracy (Ac) $\frac{TP}{TP+FN+FP}$
A		$\frac{1}{1+2} = 33\%$	$\frac{1}{1+1} = 50\%$	$\frac{1}{1+1+2} = 25\%$
B		$\frac{2}{2+0} = 100\%$	$\frac{2}{2+0} = 100\%$	$\frac{2}{2+0+0} = 100\%$
C		$\frac{0}{0+2} = 0\%$	$\frac{0}{0+2} = 0\%$	$\frac{0}{0+2+2} = 0\%$
D		$\frac{2}{2+2} = 50\%$	$\frac{2}{2+0} = 100\%$	$\frac{2}{2+0+2} = 50\%$
E		$\frac{1}{1+0} = 100\%$	$\frac{1}{1+1} = 50\%$	$\frac{1}{1+1+0} = 33.3\%$

Figure 15: Precision & Recall Metrics.

SIEMENS

Object Detection Evaluation Metrics

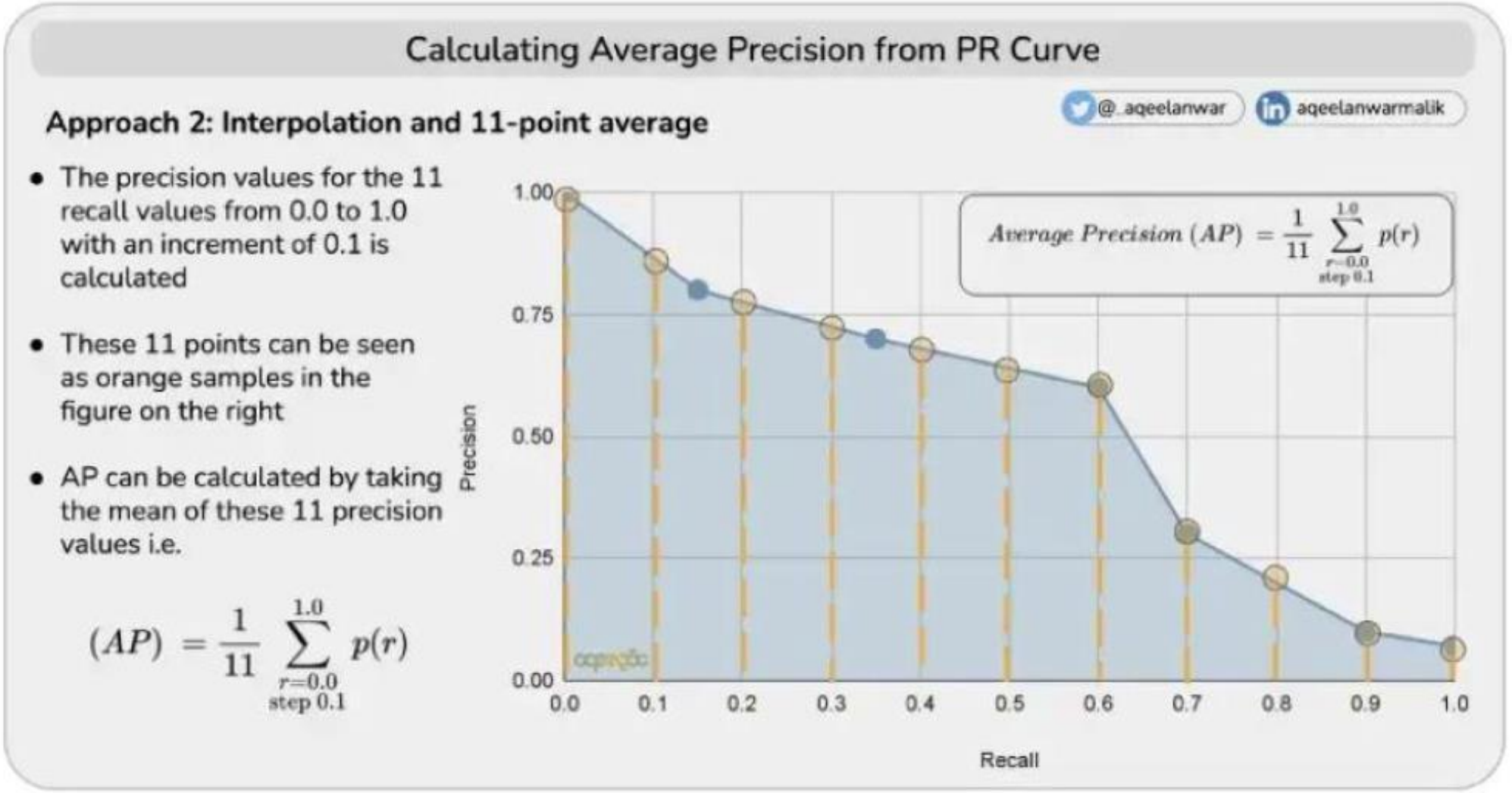


Figure 13: Average Precision from PR Curve; by interpolation and 11 point average.

Figure 16: AP at 11 Recall Points.

SIEMENS

Object Detection Evaluation Metrics: Difficulty Categorization

Algorithm 1: difficulty

Input: boxHeight,truncation,occlusion
Output: difficulty
Procedure: *Categorize Objects based on the boxHeight, truncation and occlusion thresholds adopted by KITTI*

if *boxHeight* $>= 40$ *and truncation* $<= 0.15$ *and occlusion* $\in [0]$ **then**
  $difficulty \leftarrow Easy;$
end
else if *boxHeight* $>= 40$ *and truncation* $<= 0.3$ *and occlusion* $\in [0, 1]$ **then**
  $difficulty \leftarrow Moderate;$
end
else if *boxHeight* $>= 40$ *and truncation* $<= 0.5$ *and occlusion* $\in [0, 1, 2]$ **then**
  $difficulty \leftarrow Hard;$
end
else
  $difficulty \leftarrow Ignored;$
end

Figure 17: Algorithm for assigning difficulty to ground truth object.

5 Robustness Assessment Framework

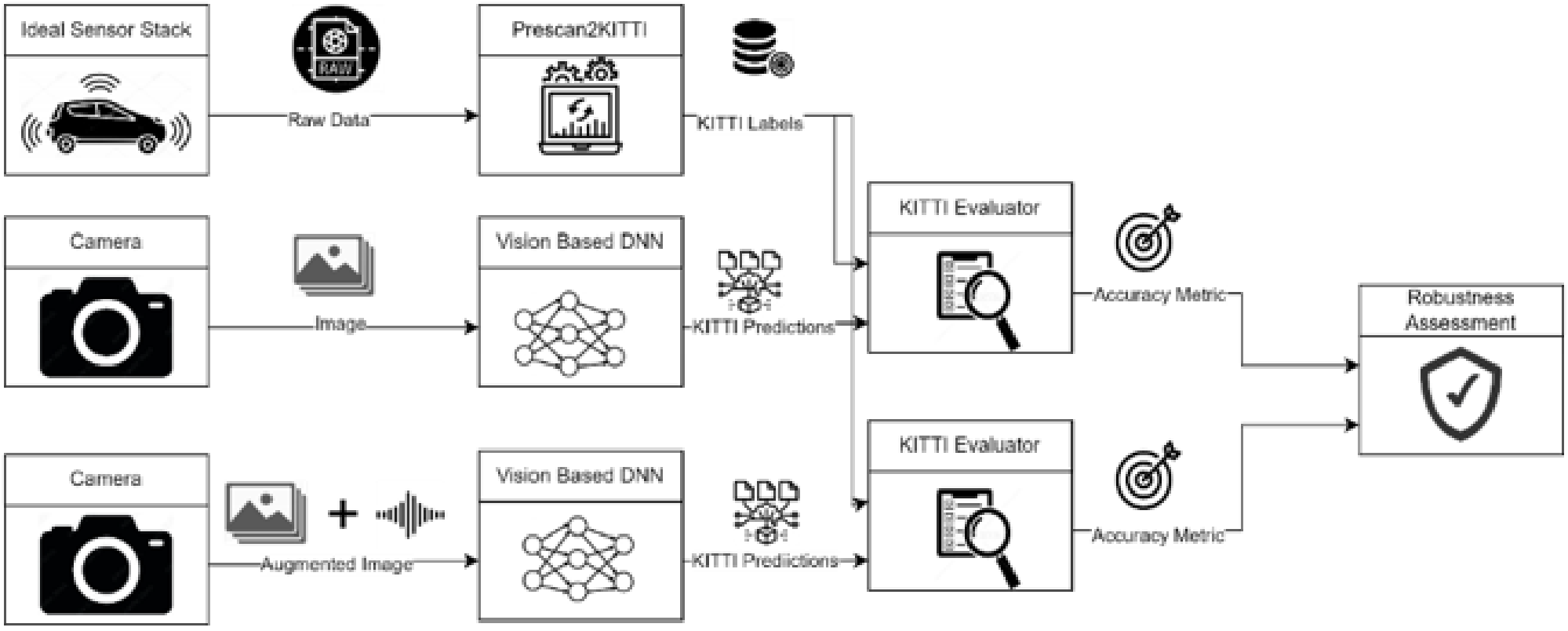


Figure 2: Framework for Robustness Assessment of Object Detectors

Figure 18: Robustness assessment requires ref dataset, mod dataset and an uncertainty metric.

SIEMENS

5 Robustness Assessment Framework: Aleatoric Uncertainties (HW Faults)

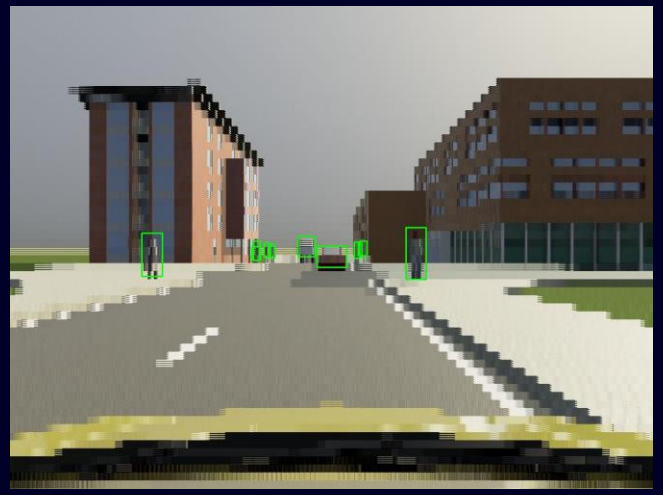


Figure 19: Row Address Fault Injection.

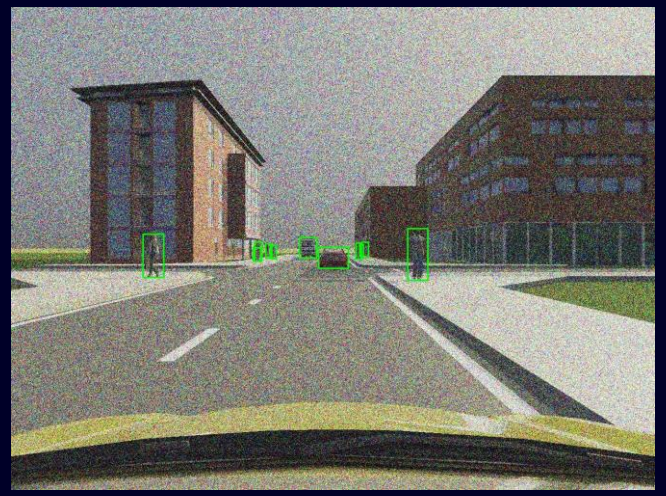


Figure 20: Random Noise Fault Injection.

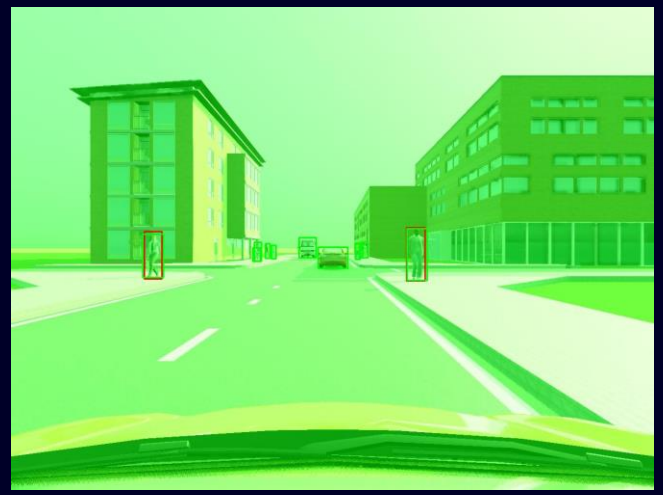


Figure 21: Color Fault Injection.

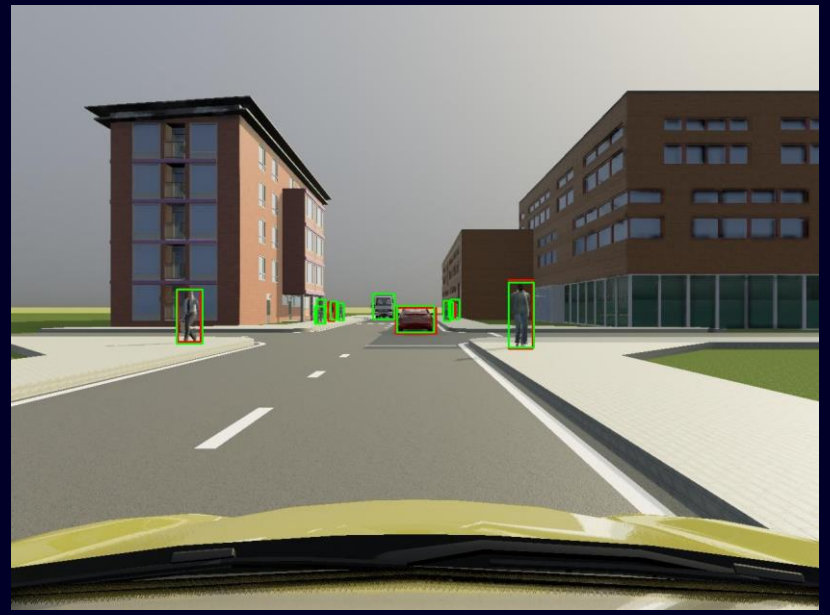


Figure 22: Reference Image

SIEMENS

5 Robustness Assessment Framework: Parametric Uncertainties

Figure 23: Fog Parametric Uncertainty.

Figure 24: Illumination Parametric Uncertainty.

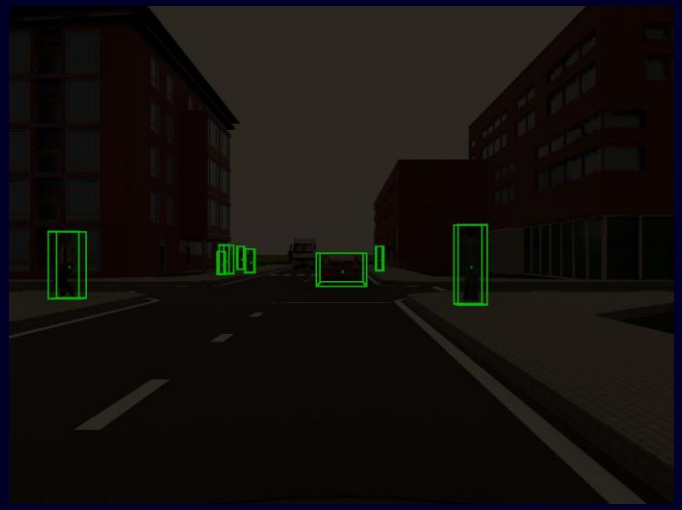


Figure 25: Illumination Parametric Uncertainty

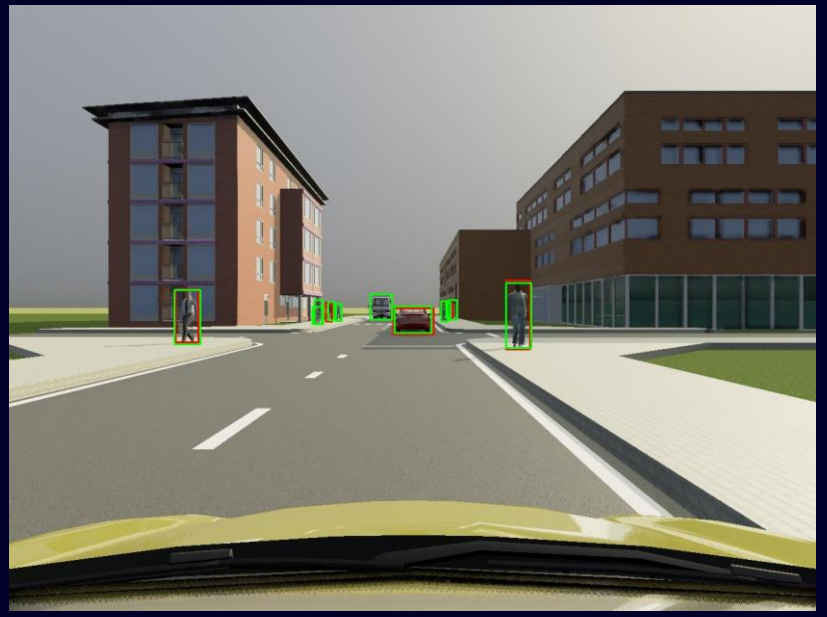


Figure 26: Reference Image

SIEMENS

5 Robustness Assessment Framework: Measuring Uncertainty

Our robustness framework refers to “A Comprehensive Evaluation Framework for Deep Model Robustness” (Jun Guoa, 2022) and uses it as guide in measuring image uncertainty.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{({\mu_x}^2 + {\mu_y}^2 + c_1)}$$

$$where\ \mu_x\ and\ \mu_y: mean\ of\ input\ images\ x\ and\ y$$

$$\sigma_x\ and\ \sigma_y: variances\ of\ input\ images\ x\ and\ y$$

$$\sigma_{xy}:\ covariance\ of\ input\ images\ x\ and\ y$$

$$c_1\ and\ c_2: are\ constants$$

Thus, Average Structural Similarity (ASS) can be defined as the average of SSIM for the complete dataset.

$$ASS = \frac{1}{m}\sum_{i=1}^{m} SSIM({x_{adv}}^{(i)}, x^{(i)})$$

$$\boldsymbol{Uncertainty\ [\mu] = 1 - ASS}$$

5 YOLOv3 Robustness Assessment against Fog

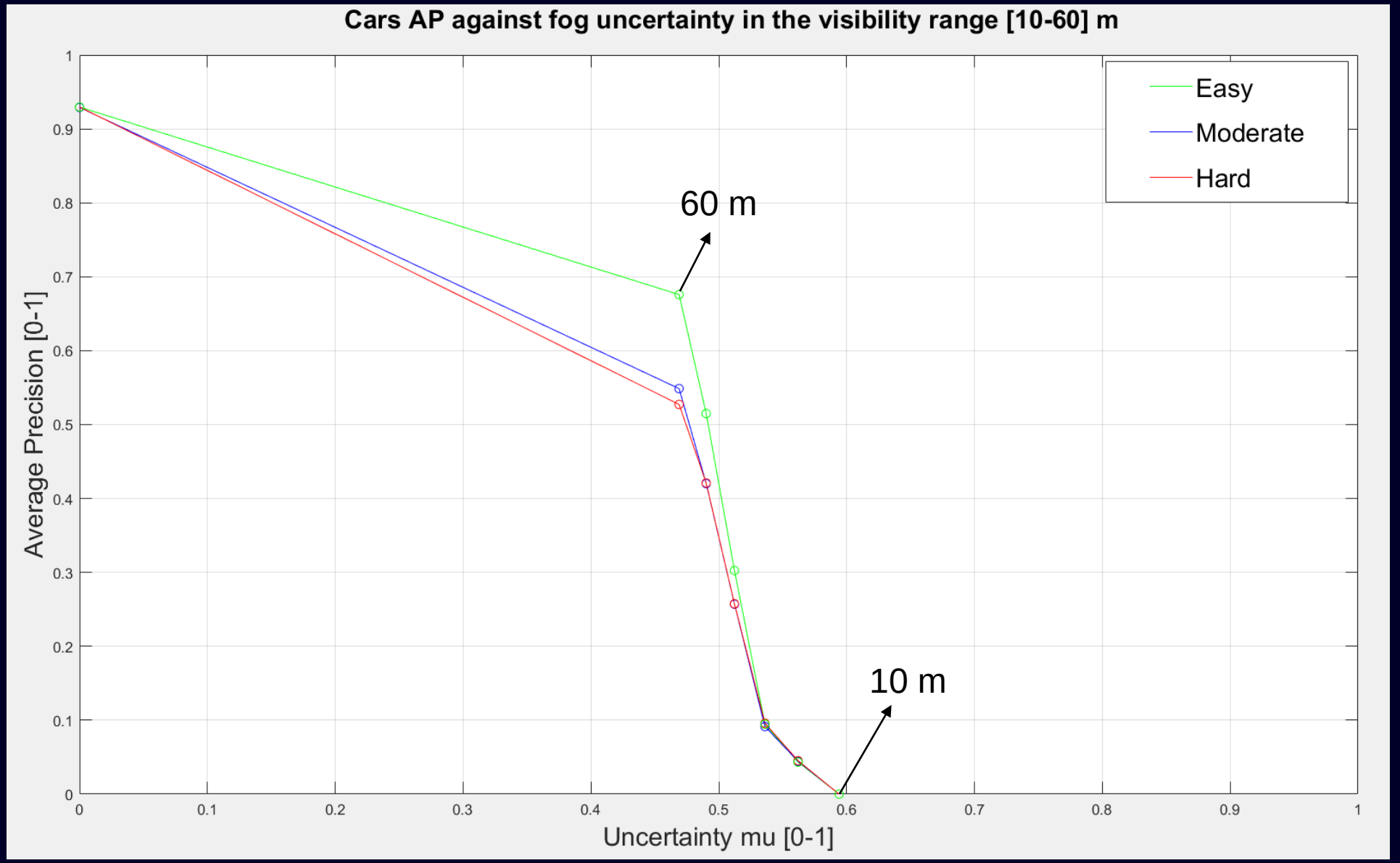


Figure 27: Performance Degradation as uncertainty increases.

SIEMENS

VRU Robustness Assessment

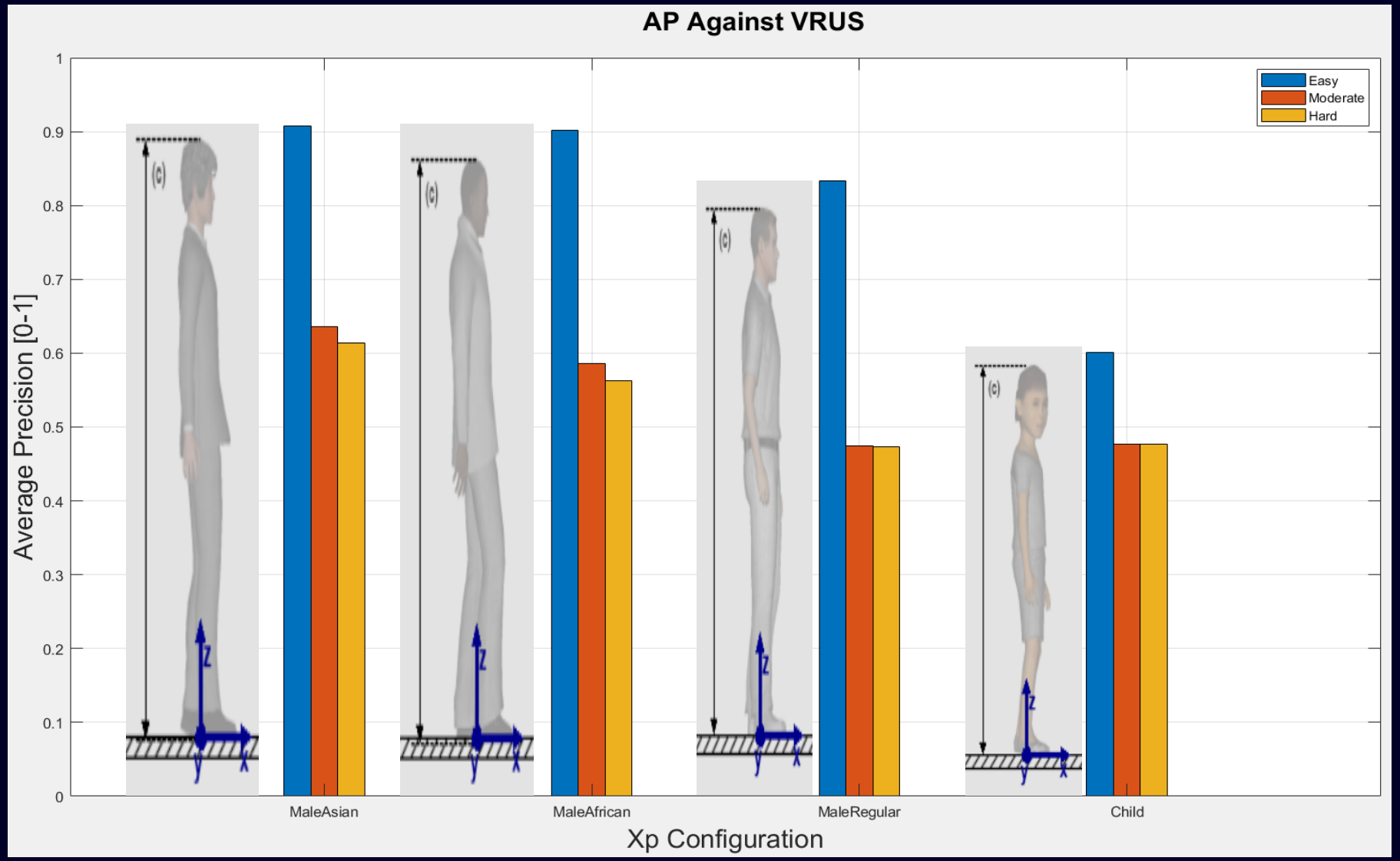


Figure 28: Comparison of performance against different VRUs.

SIEMENS

SMOKE 3D Detection

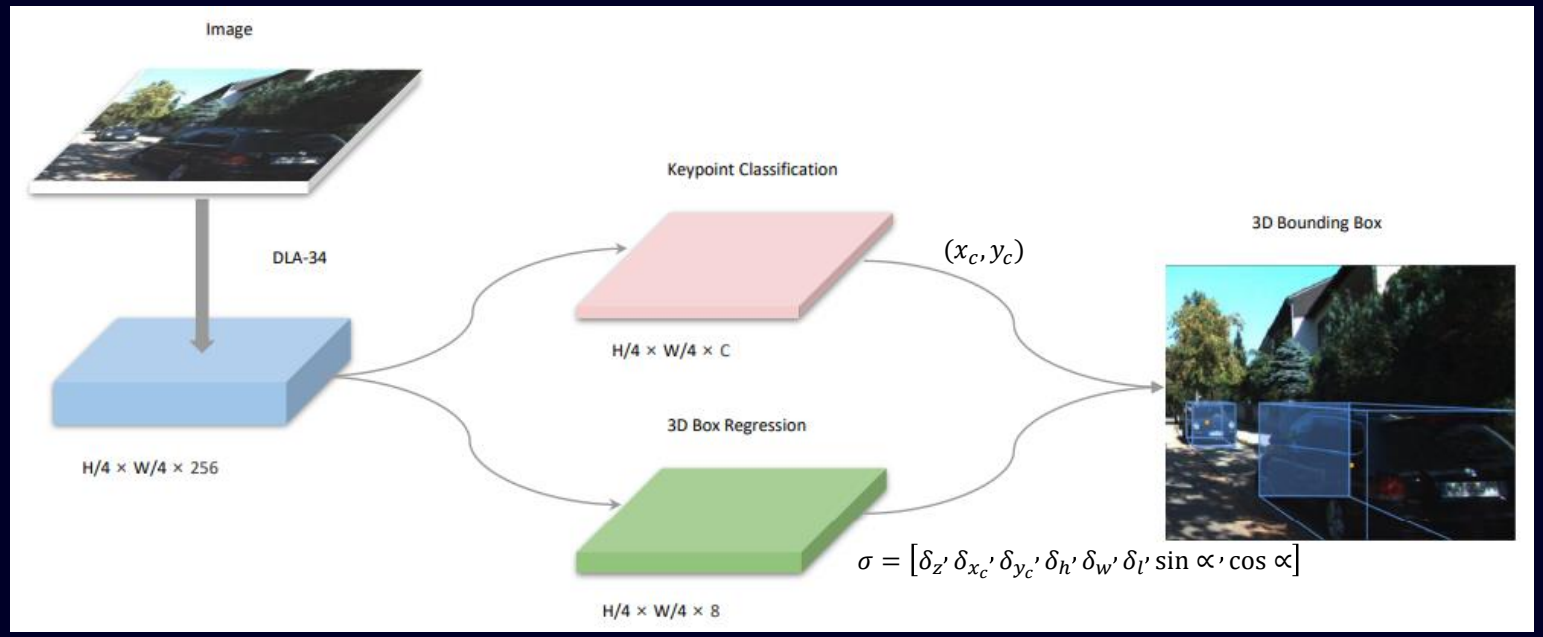


Figure 29: SMOKE Architecture.

	$Z = \sigma_z * \delta_z + \mu_z$	(3)
	$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = K^{-1} \begin{pmatrix} Z * (x_c + \delta_{x_c}) \\ Z * (y_c + \delta_{y_c}) \\ Z \end{pmatrix}$	(2)
	$\begin{pmatrix} h \\ w \\ l \end{pmatrix} = \begin{pmatrix} \bar{h} * e^{\delta_h} \\ \bar{w} * e^{\delta_w} \\ \bar{l} * e^{\delta_l} \end{pmatrix}$	(4)

Figure 30: To go from 3D center to 3D Bounding Box the shown equations are used.

Eq. (3):
- *Z = Object Depth [m]*
- *μ_z = Dataset Depth Distribution Mean*
- *σ_z = Dataset Depth Distribution STD*
- *δ_z = Depth offset [Unitless] (Predicted by SMOKE)*

Eq. (2):
- *(X,Y,Z) = 3D World Pose [m]*
- *K = Camera Matrix*
- *(x_c, y_c) = Image Plane Coordinate of Object 3D Center*

Eq. (4):
- *(h,w,l) = Predicted Object Dimensions [m]*
- *$(\bar{h}, , \bar{w}, \bar{l})$ = Mean Object Dimensions*
- *$(\delta_h, \delta_w, \delta_l)$ = Predicted Dimension Deviation From Mean*

SIEMENS

Camera Intrinsic Matrix in Prescan

The Camera Matrix to be estimated is the following:

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Where f is focal length in pixels and c optical center in pixels:

$$f_x[pixels] = \frac{img\ width\ [pixels] * f_x[mm]}{CCD\ Width\ [mm]}$$

$$f_y[pixels] = \frac{img\ height\ [pixels] * f_y[mm]}{CCD\ Height\ [mm]}$$

By setting the optical plane misalignment to 0 in x and y in Prescan:

$$c_x = \frac{img\ width\ [pixels]}{2}$$

$$c_y = \frac{img\ height\ [pixels]}{2}$$

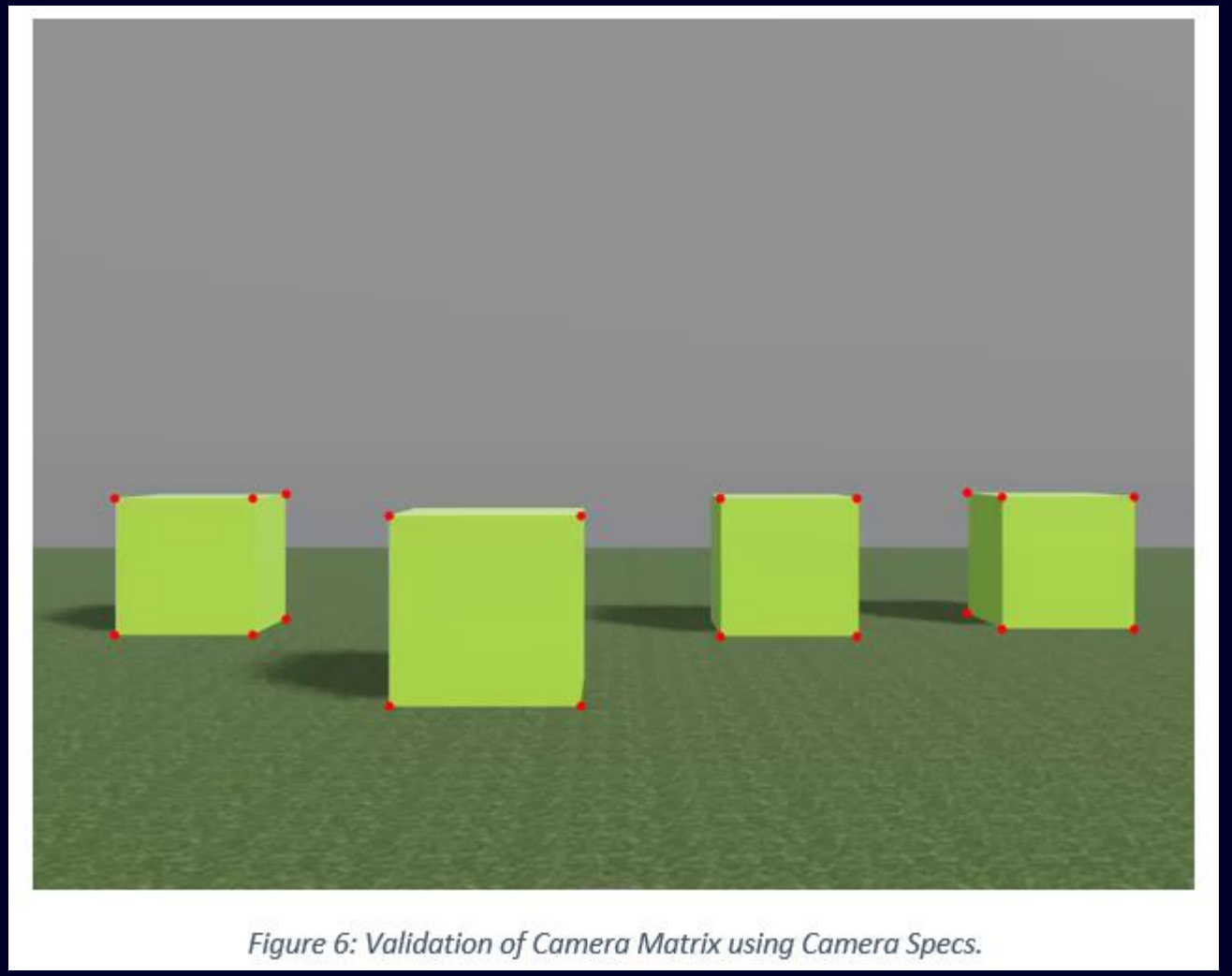


Figure 6: Validation of Camera Matrix using Camera Specs.

Figure 31: Camera Matrix Validation Experiment.

SIEMENS

6 SMOKE Depth Evaluation

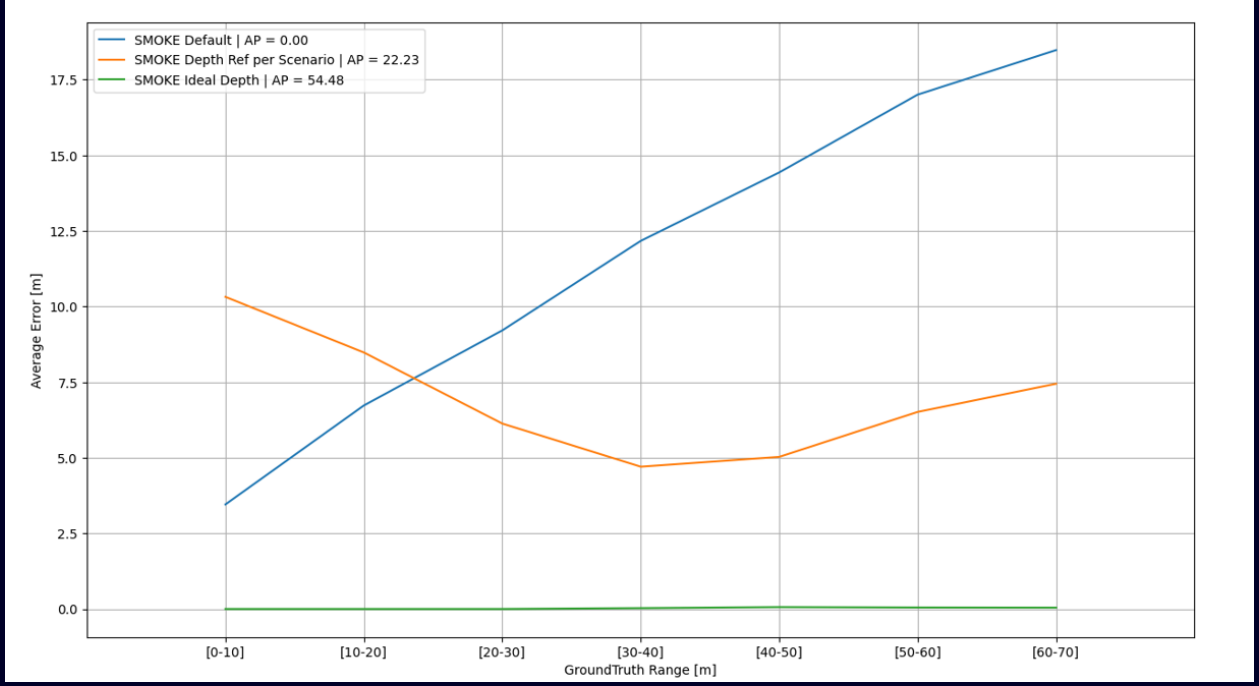


Figure 32: SMOKE Depth Evaluation on Prescan Dataset.

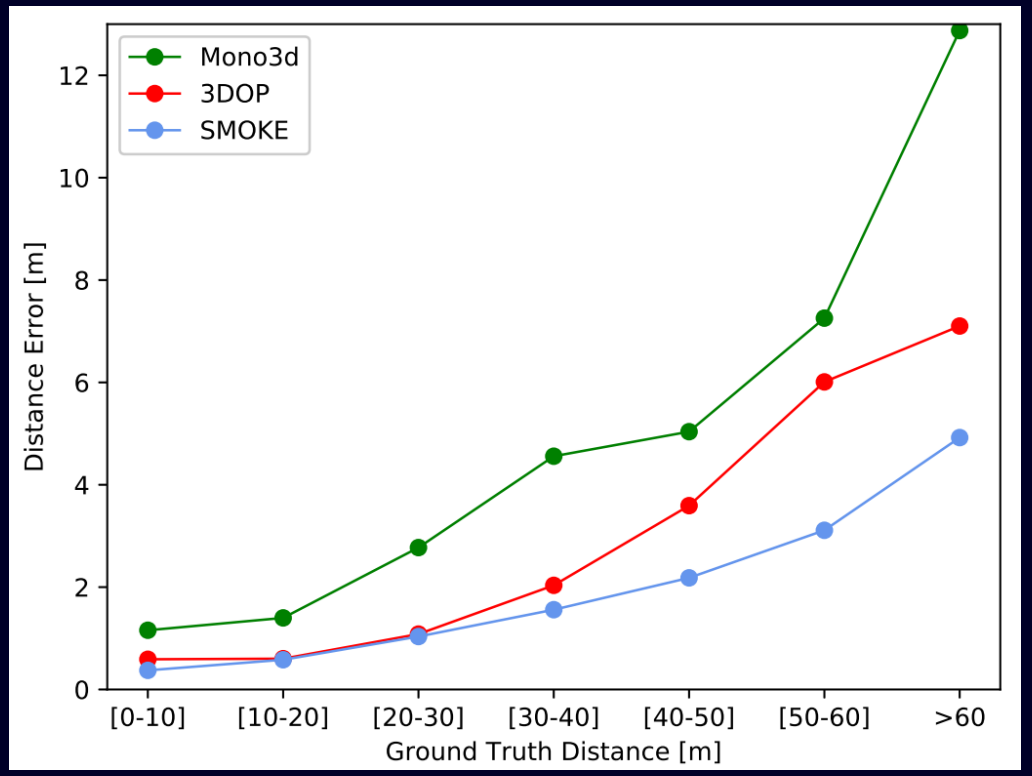


Figure 33: SMOKE Depth Evaluation on KITTI Dataset.

$$Z = \sigma_z * \delta_z + \mu_z \quad (3)$$

- *Z = Object Depth [m]*
- *μ_z = Dataset Depth Distribution Mean*
- *σ_z = Dataset Depth Distribution*
- *δ_z = Depth offset [Unitless] (Predicted by SMOKE)*

SIEMENS

6 Stereo Vision

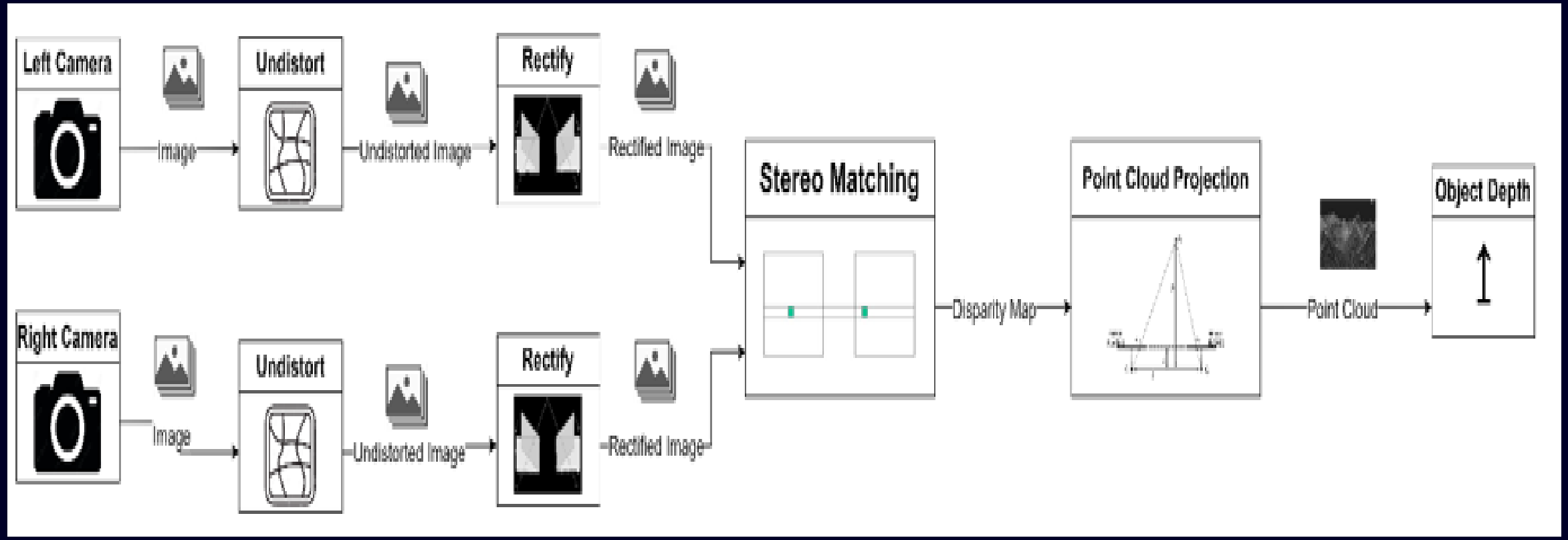


Figure 34: Our Approach To Stereo Vision.

SIEMENS

6 Stereo Rectification

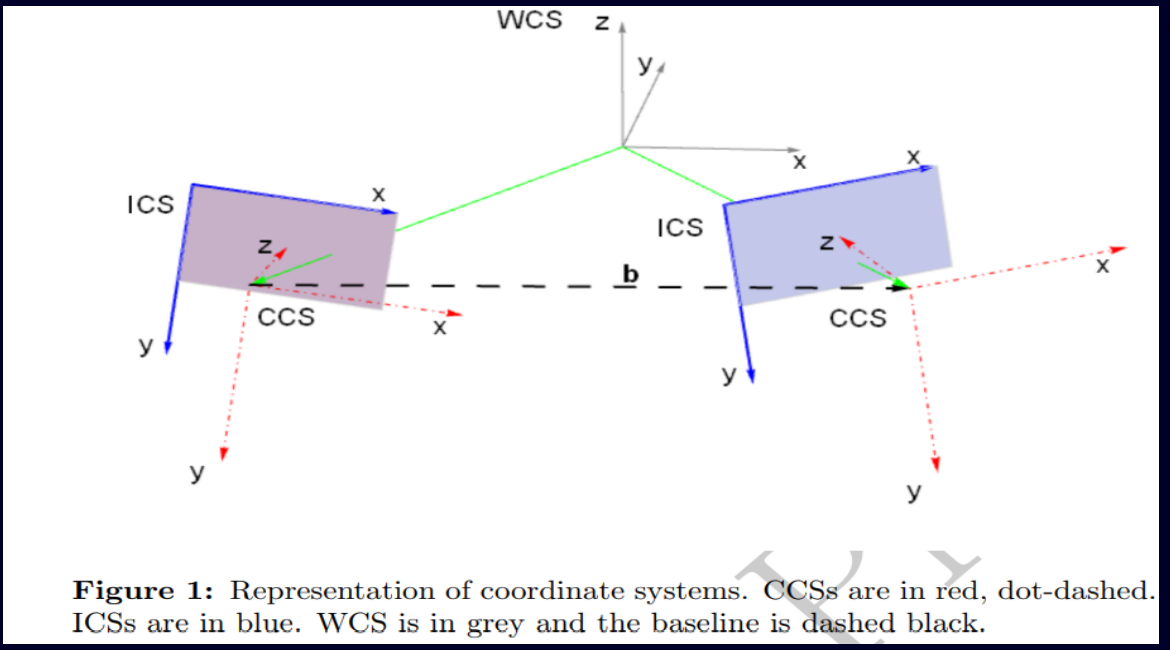


Figure 1: Representation of coordinate systems. CCSs are in red, dot-dashed. ICSs are in blue. WCS is in grey and the baseline is dashed black.

Figure 35: Unrectified Stereo Setup

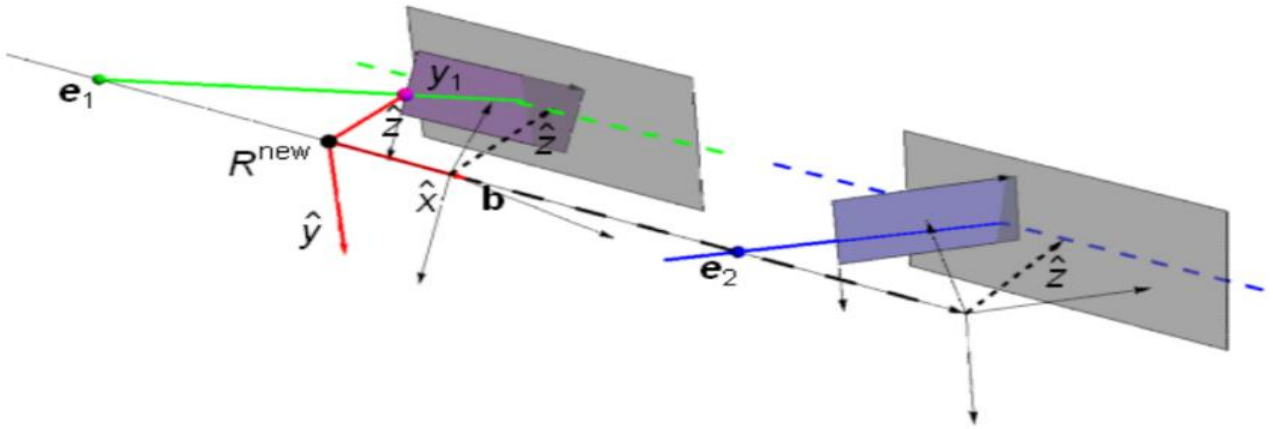


Figure 2: Common orientation of the virtual camera pair (red), projecting on a common plane (gray). $\hat{\mathbf{x}}$ is parallel to the baseline $\mathbf{b}$ (black, dashed). The corresponding epipolar lines (blue, green) and $\hat{\mathbf{z}}$ are identified by y_1 (magenta). Rectified epipolar lines are green and blue, dashed.

Figure 36: Rectified

The process of image rectification involves computing two homographies that can be applied to a pair of images to make them parallel.

A homography can be seen as a matrix or mask representing a transformation to be applied to the image.

SIEMENS

Stereo Matching

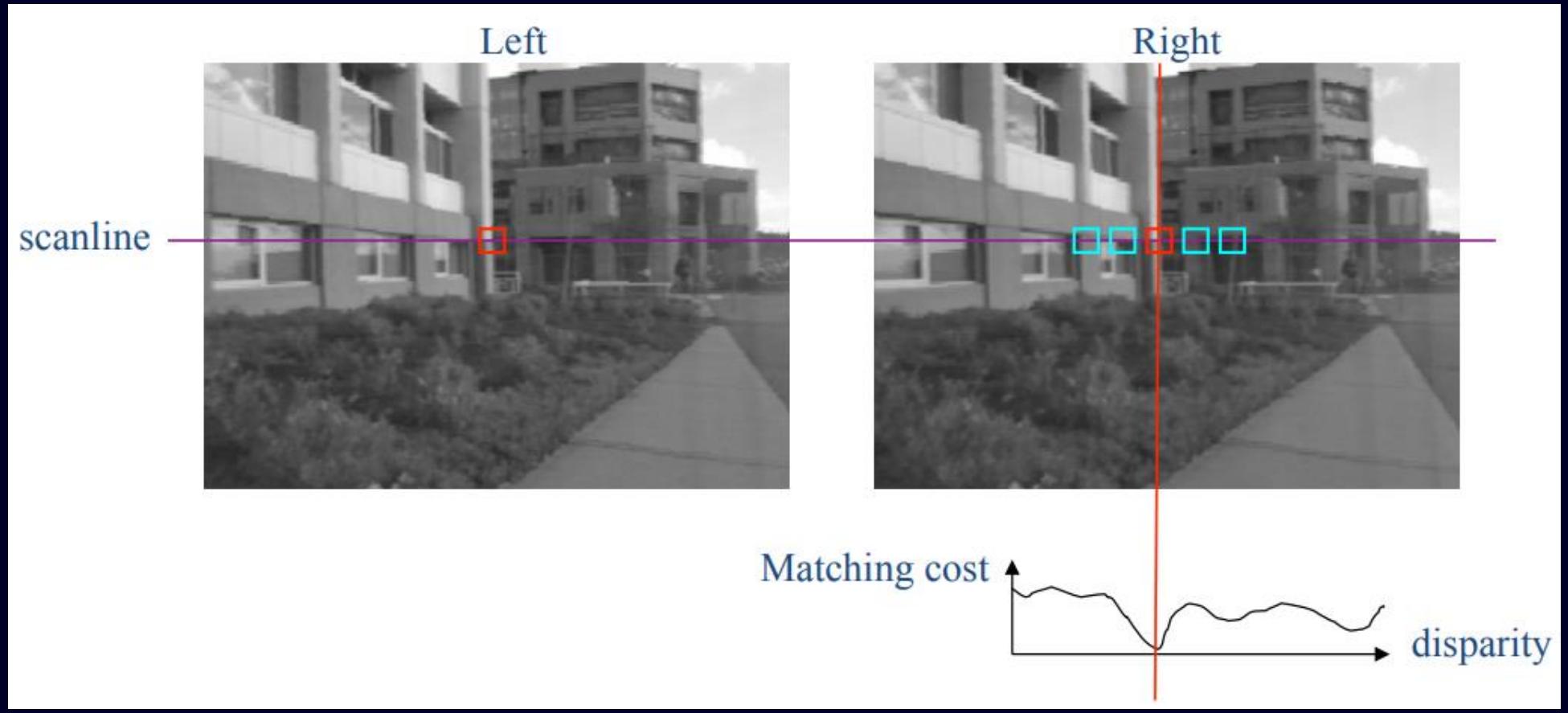


Figure 37: Stereo Matching

3D Detection State-Of-The Art

DSGN++ is an open source stereo based method which is currently state-of-the-art for 3D Detection.

$$\sigma = [\delta_x, \delta_{x_c}, \delta_{y_c}, \delta_h, \delta_w, \delta_l, \sin\alpha, \cos\alpha]$$

DSGN++: Exploiting Visual-Spatial Relation for Stereo-based 3D Detectors

Yilun Chen, *Student Member, IEEE*, Shijia Huang, *Student Member, IEEE*,
Shu Liu, *Member, IEEE*, Bei Yu, *Member, IEEE*, Jiaya Jia, *Fellow, IEEE*

https://github.com/chenyilun95/DSGN2

SIEMENS

Thank you for your attention!