Certifiably Robust Geometric Perception for Robots and Autonomous Vehicles

Luca Carlone
Assistant Professor
Massachusetts Institute of Technology

@lucacarlone1

SPARK Lab

MIT AEROASTRO

CVPR 2021 Tutorial | Adversarial Machine Learning in Computer Vision

LIDS

Certifiably Robust Geometric Perception for Robots and Autonomous Vehicles

Luca Carlone
Assistant Professor
Massachusetts Institute of Technology

@lucacarlone1

SPARK Lab

MIT AEROASTRO

CVPR 2021 Tutorial | Adversarial Machine Learning in Computer Vision

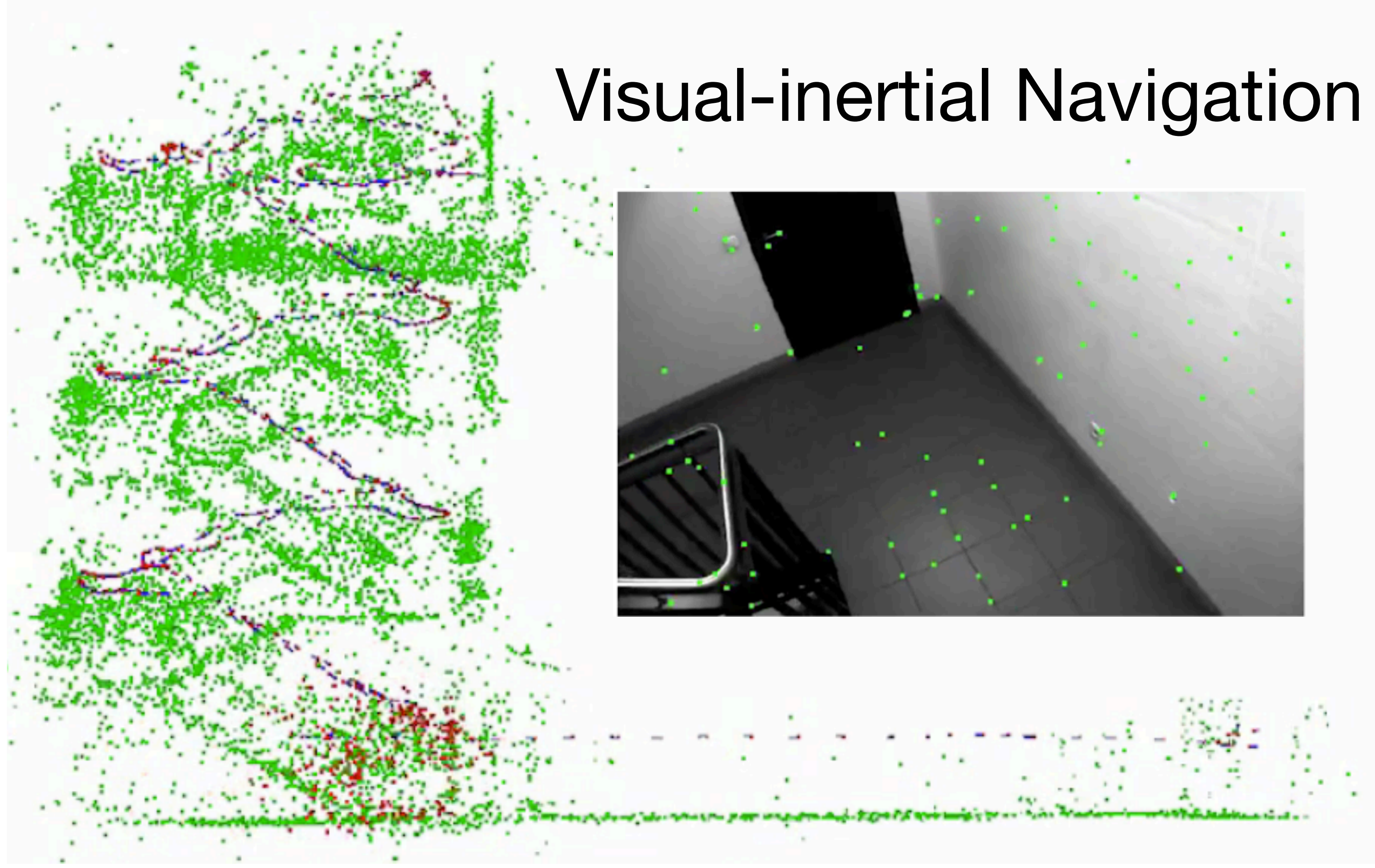
LIDS

SPARK Lab

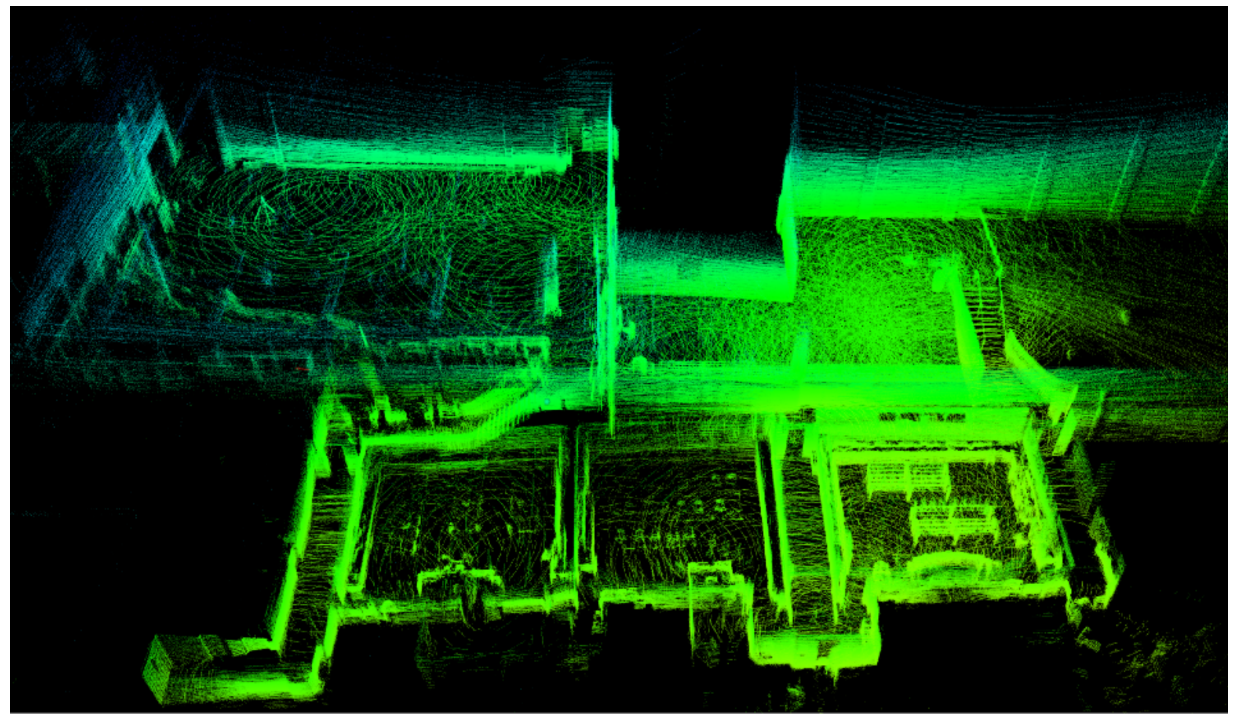
Sensing Perception Autonomy and Robot Kinetics

Robust and Certifiable Perception

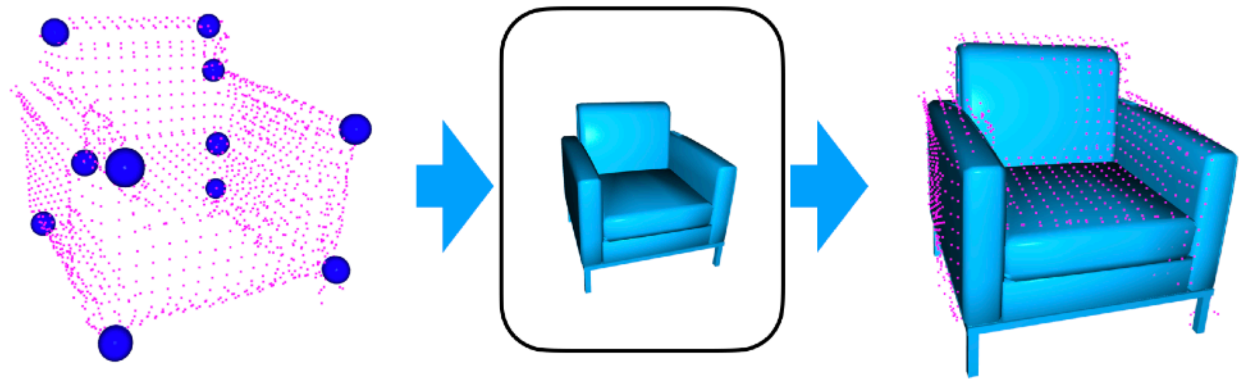
Visual-inertial Navigation



Lidar-based SLAM

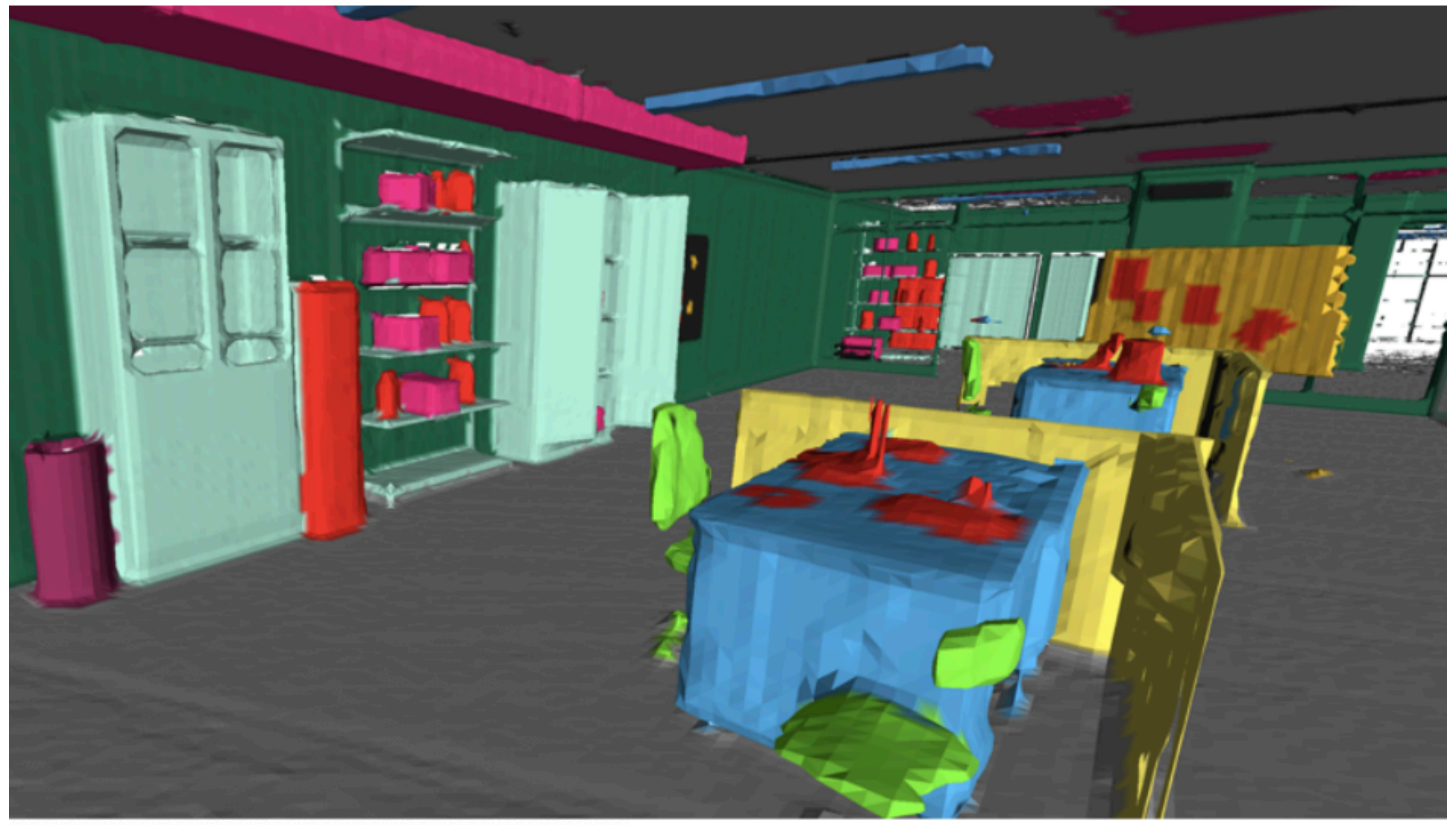


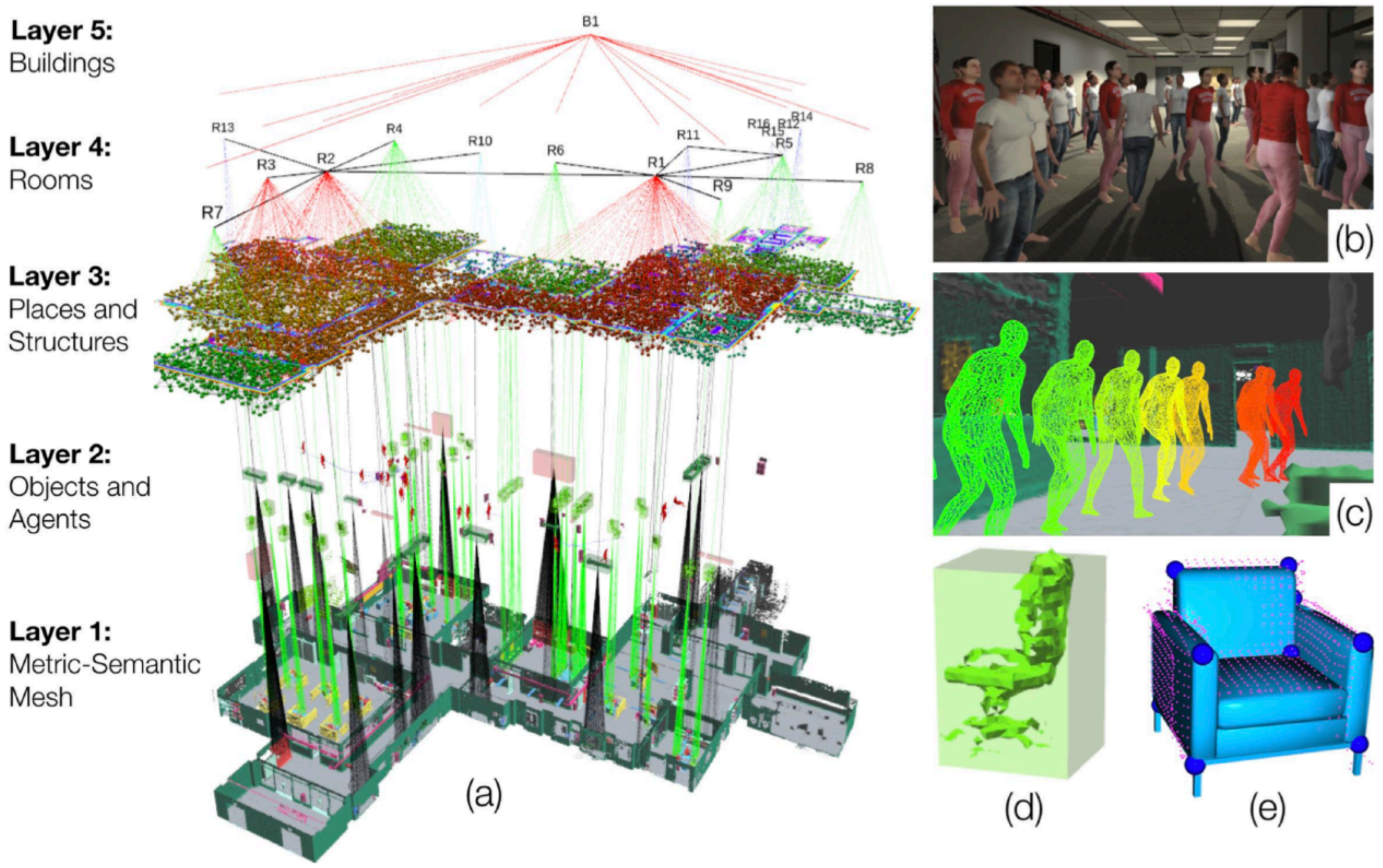
Certifiable Algorithms



High-level Scene Understanding (Spatial AI)

Kimera: Metrics-semantic SLAM

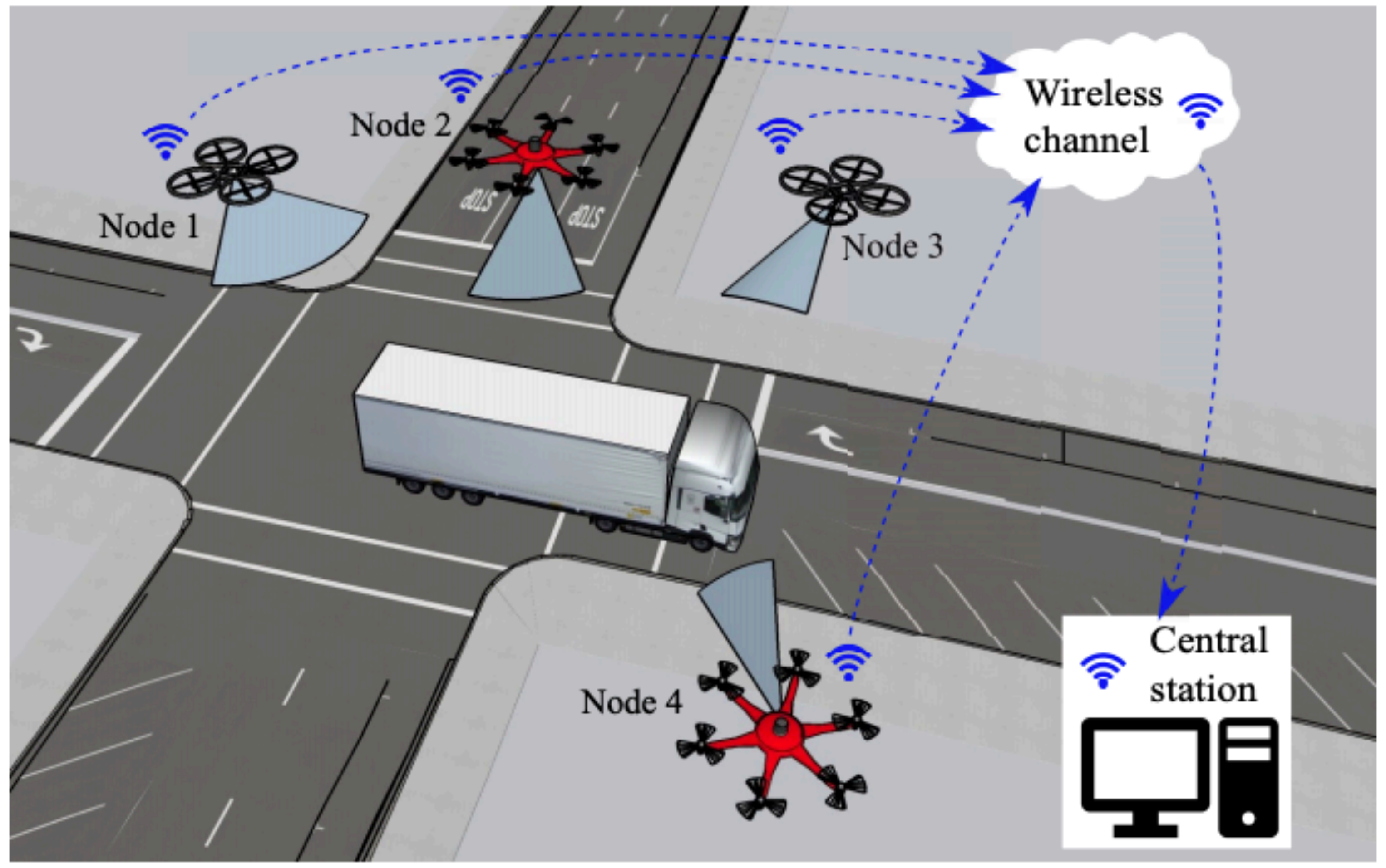


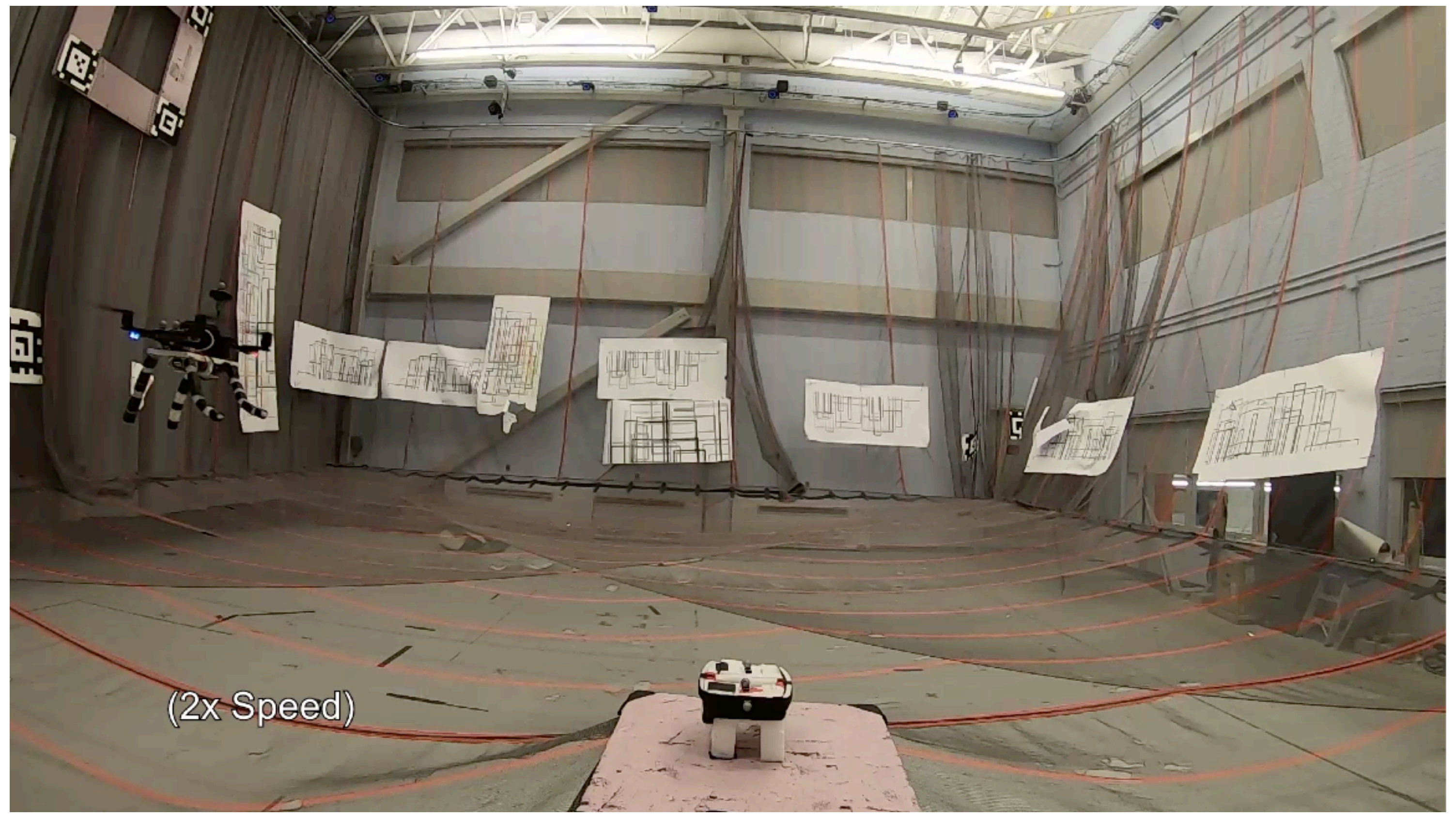


3D Dynamic Scene Graphs

Co-design

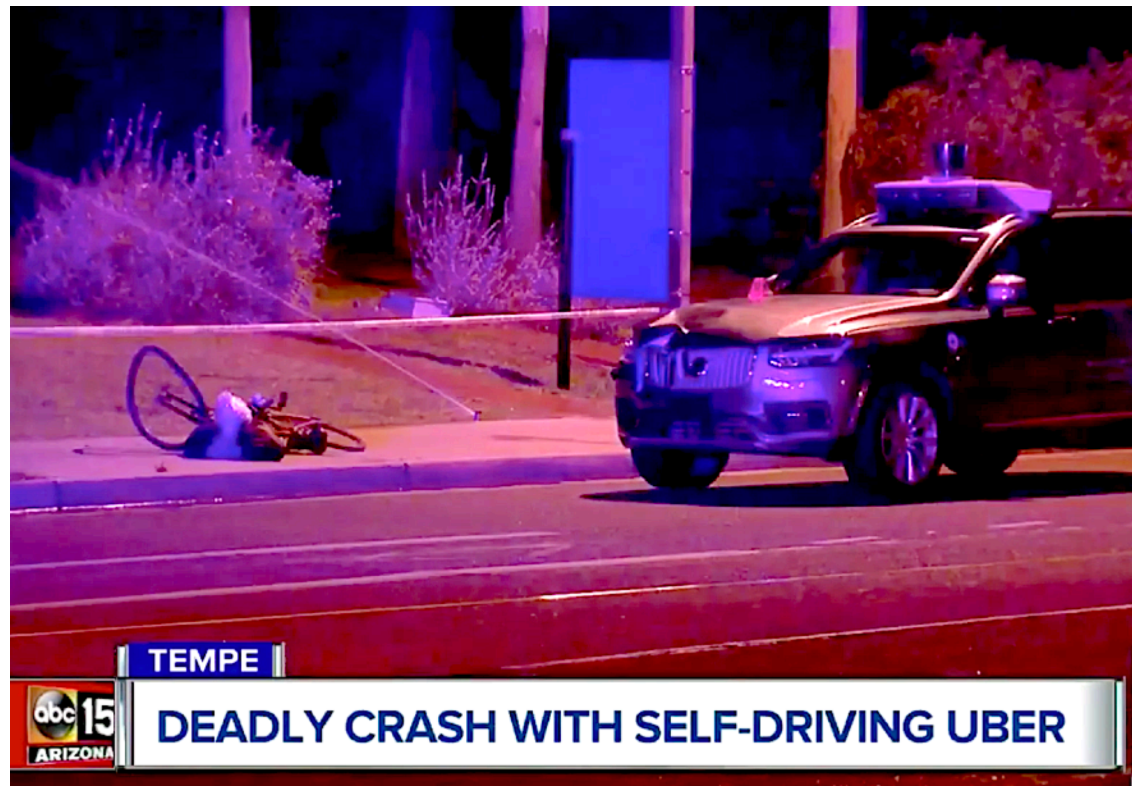
- Computation-communication co-design
- Control and sensing co-design





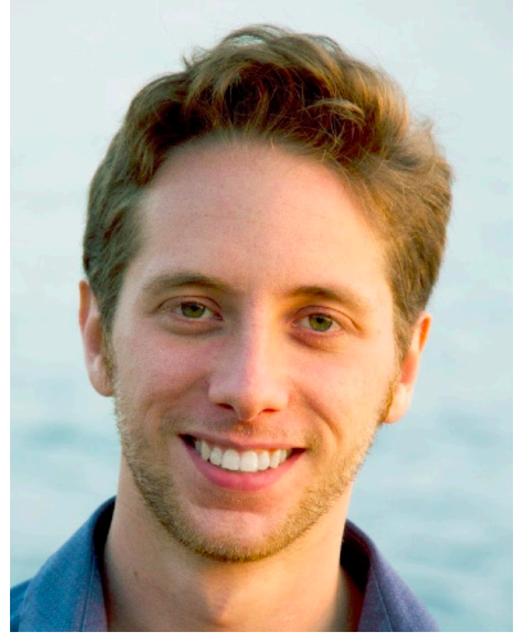
Soft Drones and Soft Aerial Manipulation

Outline

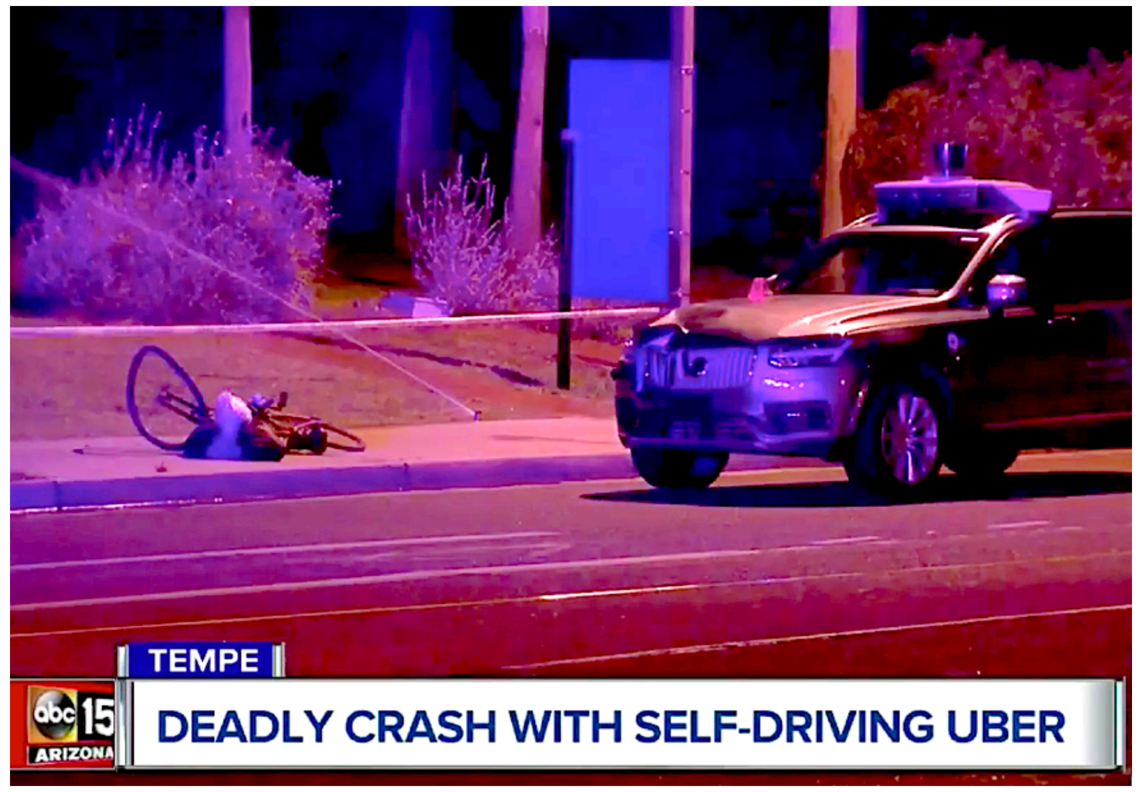


What is a “certifiable algorithm” for geometric perception?

- Graduated Non-Convexity
  [ICRA’20, CVPR’20]
- Graph-theoretic outlier pruning
  [TRO’20, ICRA’21, arxiv’21]
- Optimality certification
  [TRO’20, NeurIPS’20]



Outline



What is a "certifiable algorithm" for geometric perception?

- Graduated Non-Convexity
  [ICRA'20, CVPR'20]
- Graph-theoretic outlier pruning
  [TRO'20, ICRA'21, arxiv'21]
- Optimality certification
  [TRO'20, NeurIPS'20]

Geometric perception example: image-based object localization

RGB images: [Gu&Kanade, CVPR’06][Lin, ECCV’14][Zhou, CVPR’15][Pavlakos, ICRA’17][Xinke, RSS’19][Yang, CVPR’20]
Point clouds: [PointNetLK, CVPR’19][DCP, ICCV’19][SmoothNet, CVPR’19][TEASER, RSS’19, TRO’20]

(Generality)

Perception Front-end
Perception Back-end
Feature detection & description
Model fitting/ Estimation
Object localization in images
Feature detection & description
3D registration
Object localization in point clouds
SLAM (visual-inertial navigation, Structure from Motion)
Camera images
Inertial Measurements
Sensor Front-end
Factor Graph Optimization

Image-based object localization: perception issues

***ISSUE 1**: front-end (hand-crafted or deep-learned) can produce many misdetections (not uncommon to have >90% outliers)*

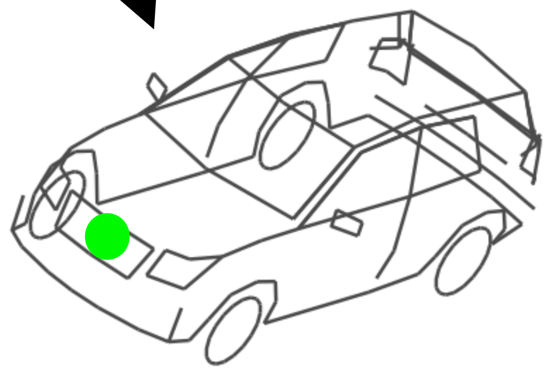
***ISSUE 2**: back-end may fail if there are many outliers*

RGB images: [Gu&Kanade, CVPR'06][Lin, ECCV'14][Zhou, CVPR'15][Pavlakos, ICRA'17][Xinke, RSS'19][Yang, CVPR'20]
Point clouds: [PointNetLK, CVPR'19][DCP, ICCV'19][SmoothNet, CVPR'19][TEASER, RSS'19, TRO'20]

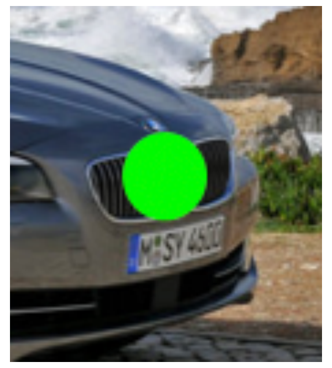
Why does the back-end fail?

$$\min_{\boldsymbol{x} \in \mathbb{X}} \sum_{i \in \mathcal{M}} \| r(\boldsymbol{x}, \boldsymbol{y}_i) \|^2$$

Set of measurements Residual



Estimate



Measurement "i"

[Dellaert and Kaess, Factor Graphs for Robot Perception, FnT 2017]
[Barfoot. State Estimation for Robotics. Cambridge University Press 2017]

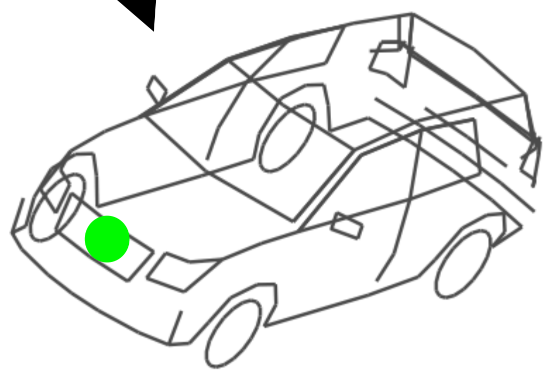
Why does the back-end fail?

Loss function

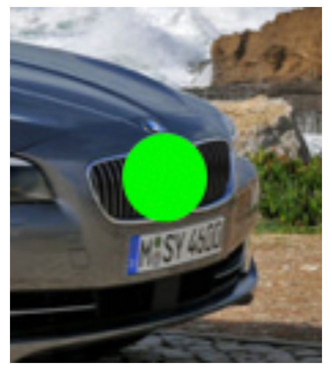
$$\min_{\boldsymbol{x}\in\mathbb{X}} \sum_{i\in\mathcal{M}} \rho(r(\boldsymbol{x}, \boldsymbol{y}_i))$$

Set of measurements

Residual



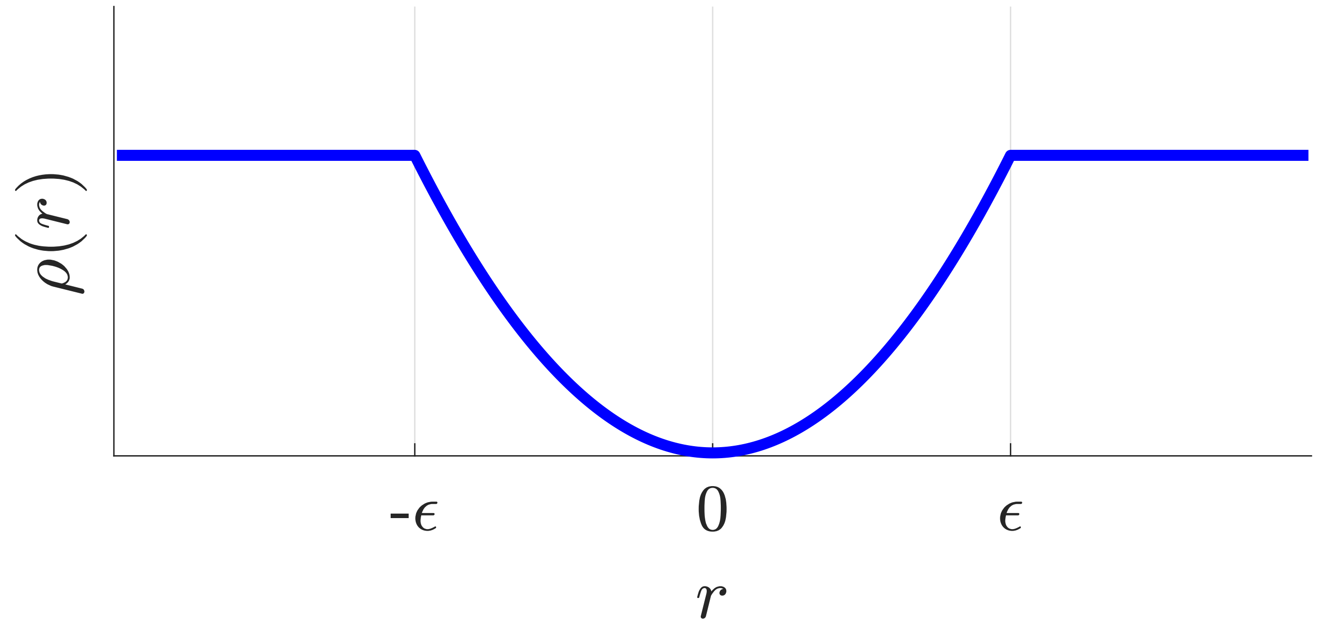
Estimate



Measurement “i”

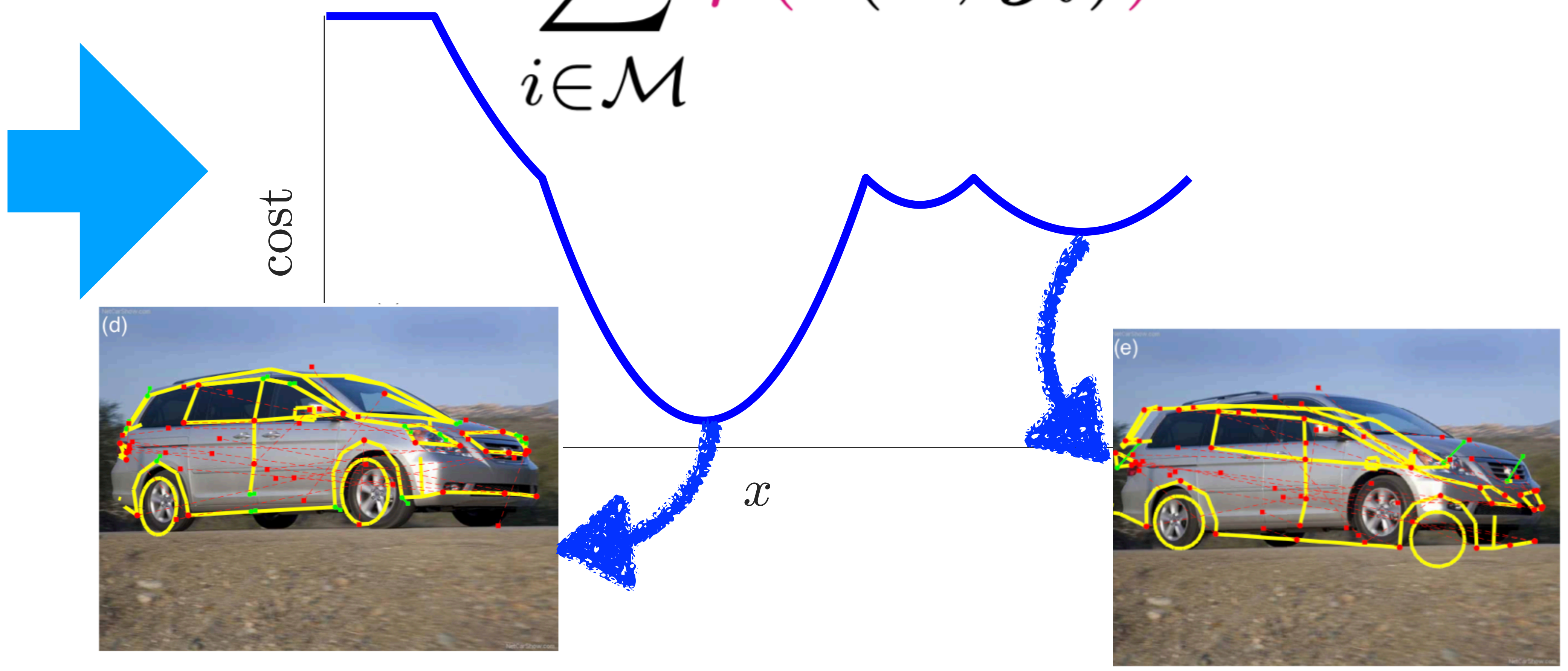
ISSUE: for common choices of loss function ρ, the problem is non-convex:

$$\rho(r(\boldsymbol{x}, \boldsymbol{y}_i))$$



Truncated Least Squares Loss

$$\sum_{i\in\mathcal{M}} \rho(r(\boldsymbol{x}, \boldsymbol{y}_i))$$



State of the art:
- **Local solvers:** need initial guess, stuck in local minima
- **RANSAC**: fails with many outliers, low-dimensional problems, non-deterministic
- both fail without notice

Our goal: Certifiable Algorithms

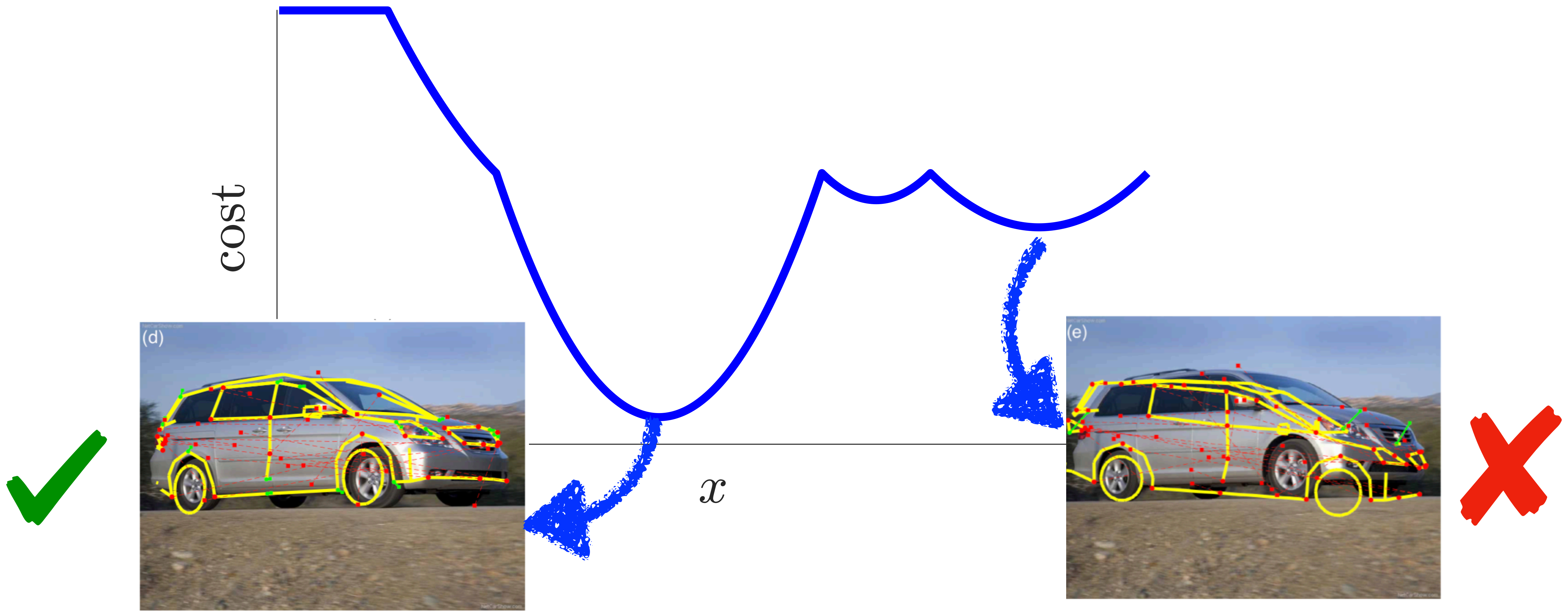
Input
(measurements y_i)

Algorithm

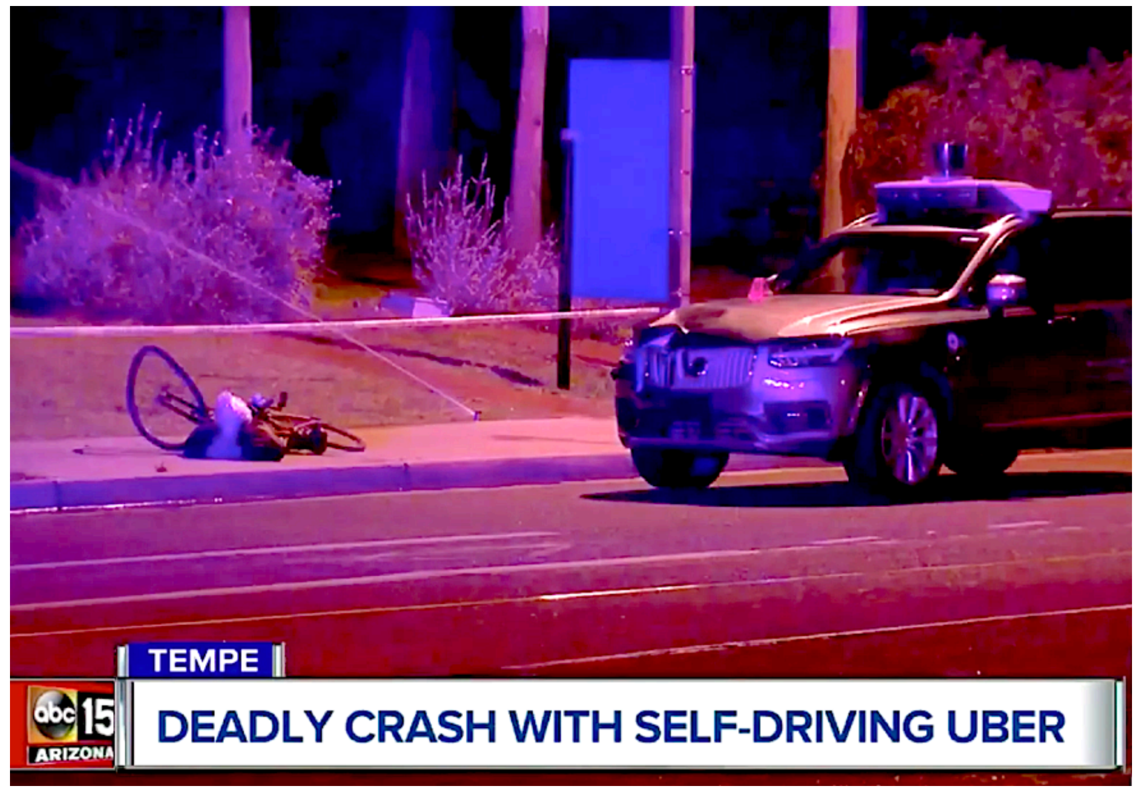
$$\min_{\boldsymbol{x} \in \mathbb{X}} \sum_{i \in \mathcal{M}} \rho(r(\boldsymbol{x}, \boldsymbol{y}_i))$$

Output
(estimate)

Certifiable algorithms: fast algorithms that (<u>w/o requiring an initial guess</u>) solve robust estimation to <u>optimality</u> in virtually all problem instances <u>or detect failure</u> when unable to compute an optimal solution



Outline



What is a "certifiable algorithm" for geometric perception?

- Graduated Non-Convexity
  [ICRA'20, CVPR'20]
- Graph-theoretic outlier pruning
  [TRO'20, ICRA'21, arxiv'21]
- Optimality certification
  [TRO'20, NeurIPS'20]

Black-Rangarajan Duality

Theorem 1 [Informal - Black, Rangarajan, 1996] We can rewrite common robust loss functions by adding auxiliary variables θ_i (one for each measurement)

$$\arg\min_{x\in\mathbb{X}} \sum_{i\in\mathcal{M}} \rho\left(r_i(\boldsymbol{x}, \boldsymbol{y}_i)\right)$$

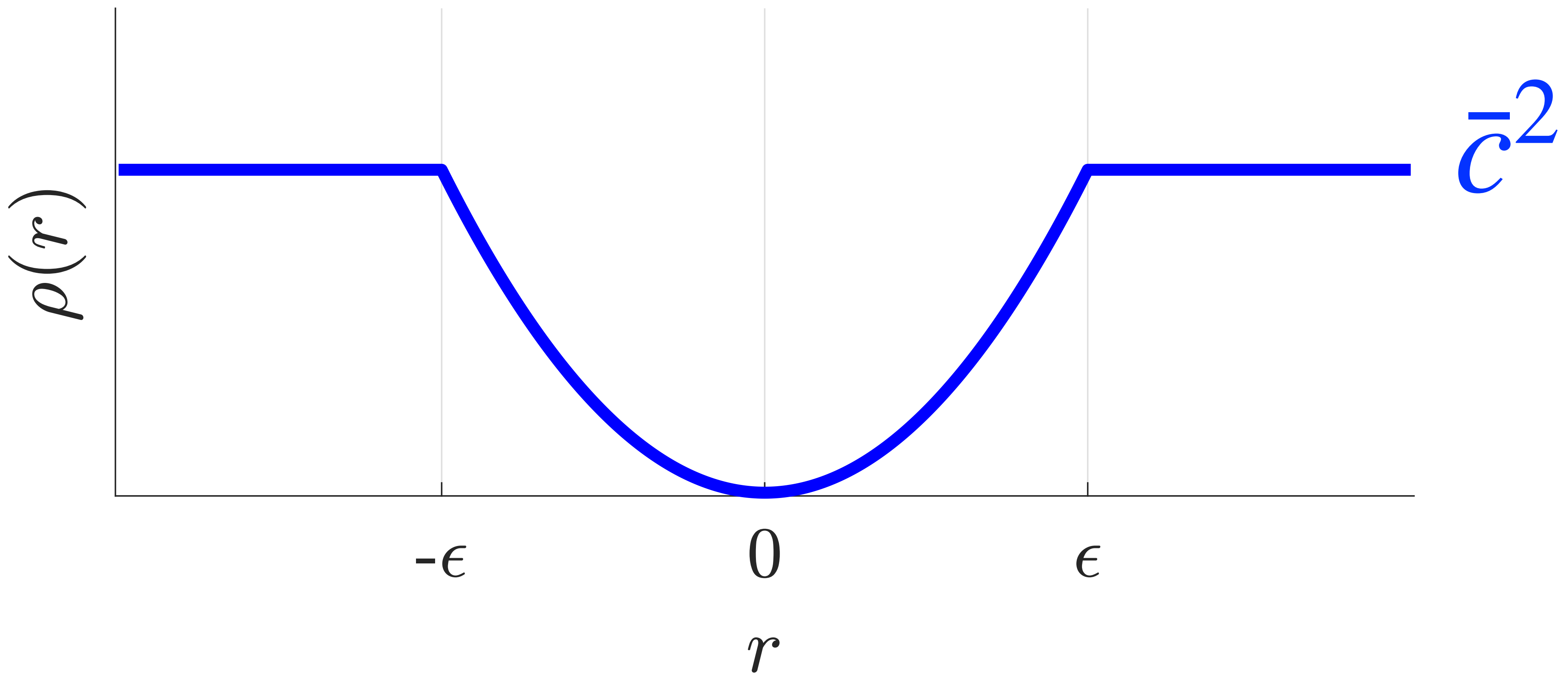
Robust loss function

$$\Updownarrow$$

$$\arg\min_{x\in\mathbb{X},\theta_i\in[0,1]} \sum_{i\in\mathcal{M}} \theta_i r_i^2(\boldsymbol{x}, \boldsymbol{y}_i) + \Phi_\rho(\theta_i)$$

"Outlier process"

Example: Truncated Least Squares (TLS)



$$\arg\min_{x\in\mathbb{X}} \sum_{i\in\mathcal{M}} \min\left(\|r_i(x, y_i)\|^2 \; , \; \bar{c}^2\right)$$

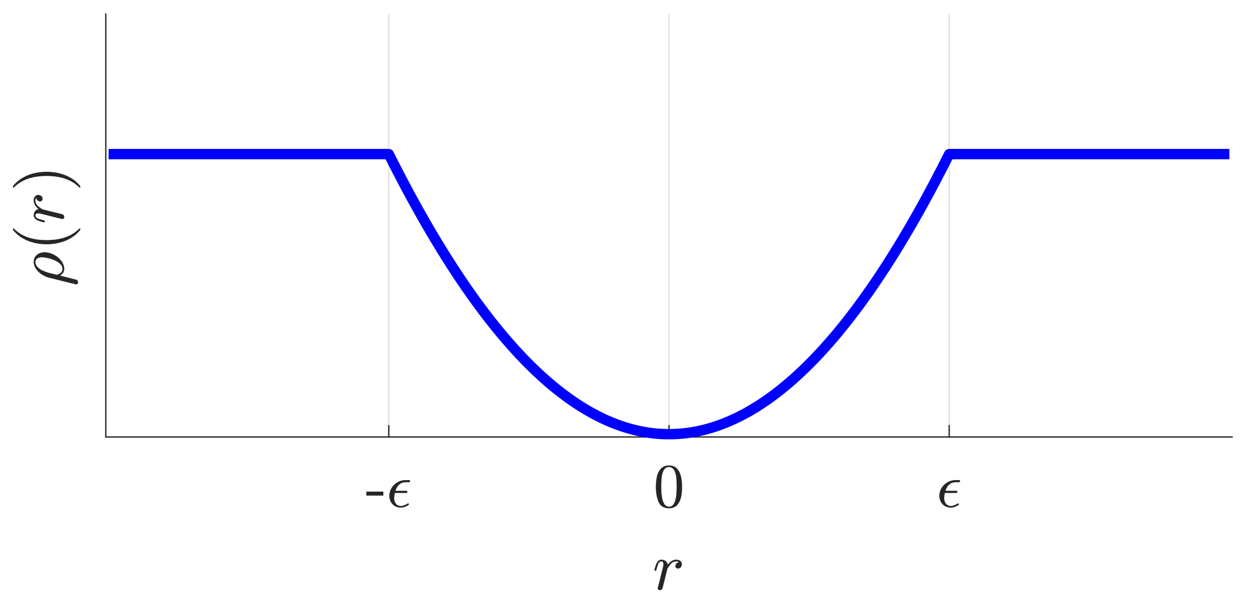
$$\Updownarrow$$

$$\arg\min_{\substack{x\in\mathbb{X},\\ \theta_i\in\{0,1\},\forall i}} \sum_{i\in\mathcal{M}} \theta_i \|r_i(x, y_i)\|^2 + (1-\theta_i)\bar{c}^2$$

[Black, Rangarajan, "On the Unification of Line Processes, Outlier Rejection, and Robust Statistics with Applications in Early Vision, IJCV'96.]

First attempt: Alternating Minimization

$$\arg\min_{\substack{x \in \mathbb{X}, \\ \theta_i \in \{0,1\}, \forall i}} \sum_{i \in \mathcal{M}} \theta_i \| r_i(x, y_i) \|^2 + (1 - \theta_i)\bar{c}^2$$



Potential approach: Alternating Minimization (Block Coordinate Descent)

a **Variable Update:** fix weights θ_i, optimize variable x

✓ becomes a weighted least squares problem

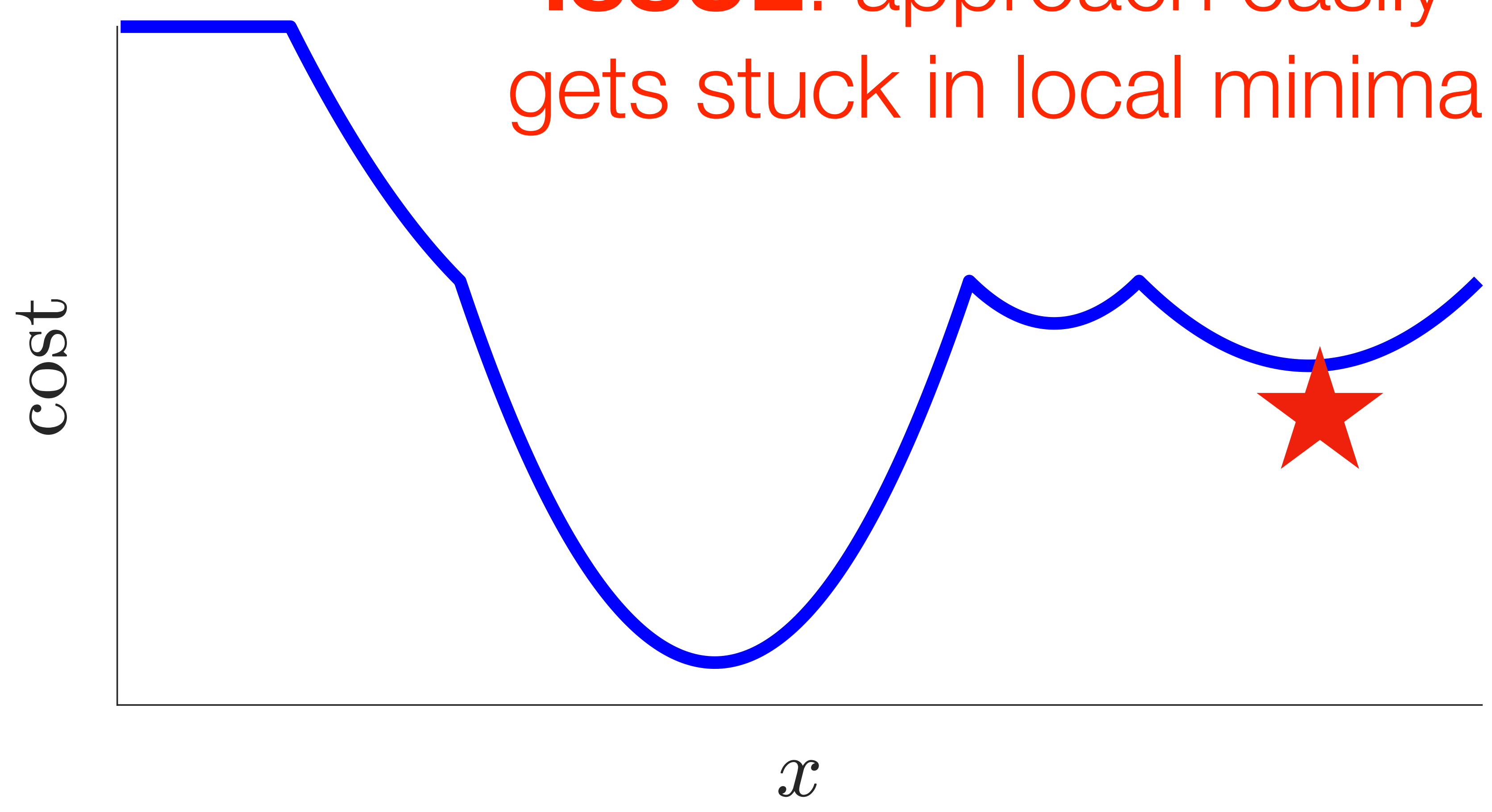
✓ can be solved globally using non-minimal solvers

b **Weight Update:** fix variable x, optimize weights θ_i

✓ splits into scalar optimization problems

✓ can be solved in closed form

ISSUE: approach easily gets stuck in local minima



[https://mit-spark.github.io/GlobalOptimization-ICCV2019/]

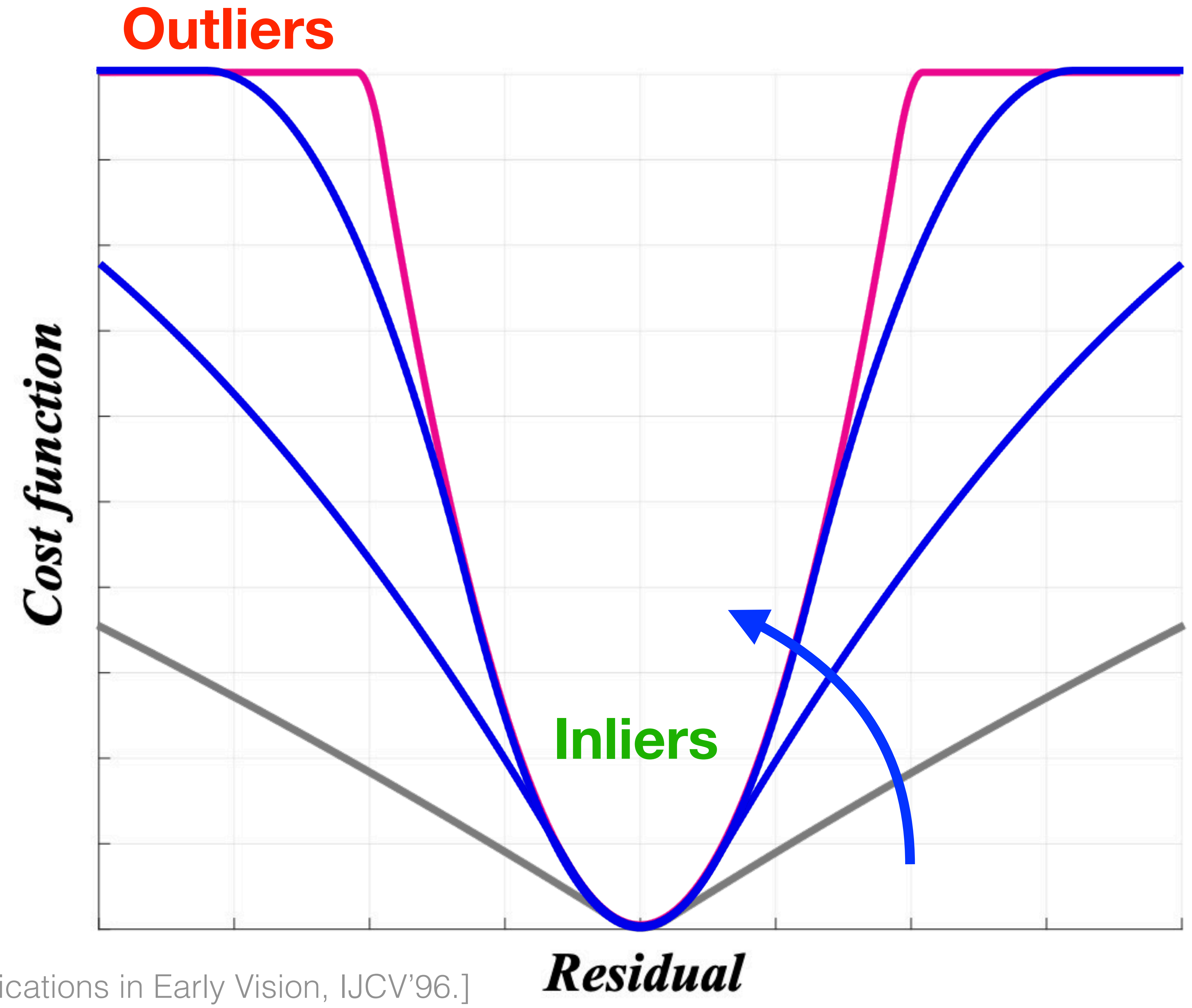
Teaching an old dog new tricks: Graduated Non-Convexity (GNC)

$$\underset{\substack{x \in \mathbb{X}, \\ \theta_i \in \{0,1\}, \forall i}}{\arg\min} \sum_{i \in \mathcal{M}} \theta_i \|r_i(x, y_i)\|^2 + (1 - \theta_i)\bar{c}^2$$

Issue: alternation scheme is trapped in local minima due to non-convexity

Key idea:

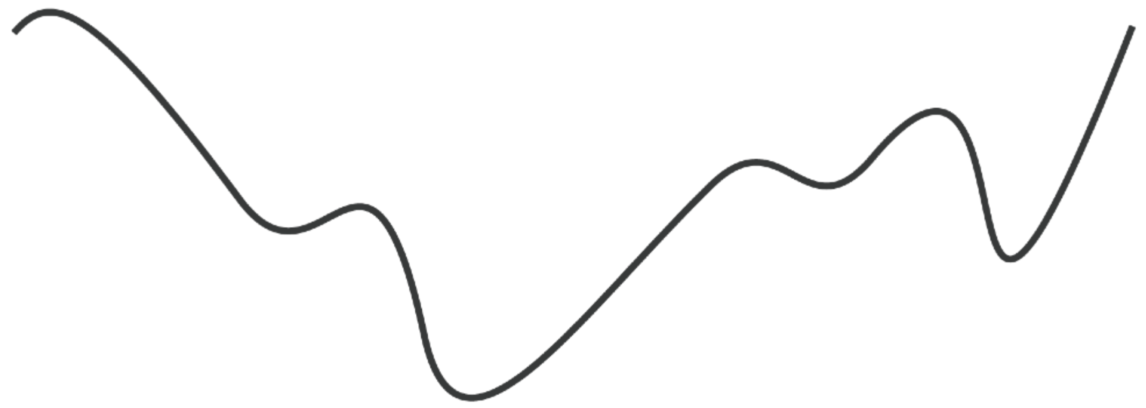
- start from a convex approximation of the cost function
- gradually increase non-convexity until you recover original function
- [proposed for early vision problems by Black & Rangarajan]



[Black, Rangarajan, "On the Unification of Line Processes, Outlier Rejection, and Robust Statistics with Applications in Early Vision, IJCV'96.]
[Yang, Antonante, Tzoumas, Carlone. Graduated non-convexity for robust spatial perception: from non-minimal solvers to global outlier rejection. RAL 2020. (best paper in robot vision at ICRA 2020)]

Graduated Non-Convexity (GNC): Algorithm

$$\boxed{\arg\min_{\substack{x \in \mathbb{X}, \\ \theta_i \in \{0,1\}, \forall i}} \sum_{i \in \mathcal{M}} \theta_i \| r_i(x, y_i) \|^2 + (1 - \theta_i)\bar{c}^2}$$

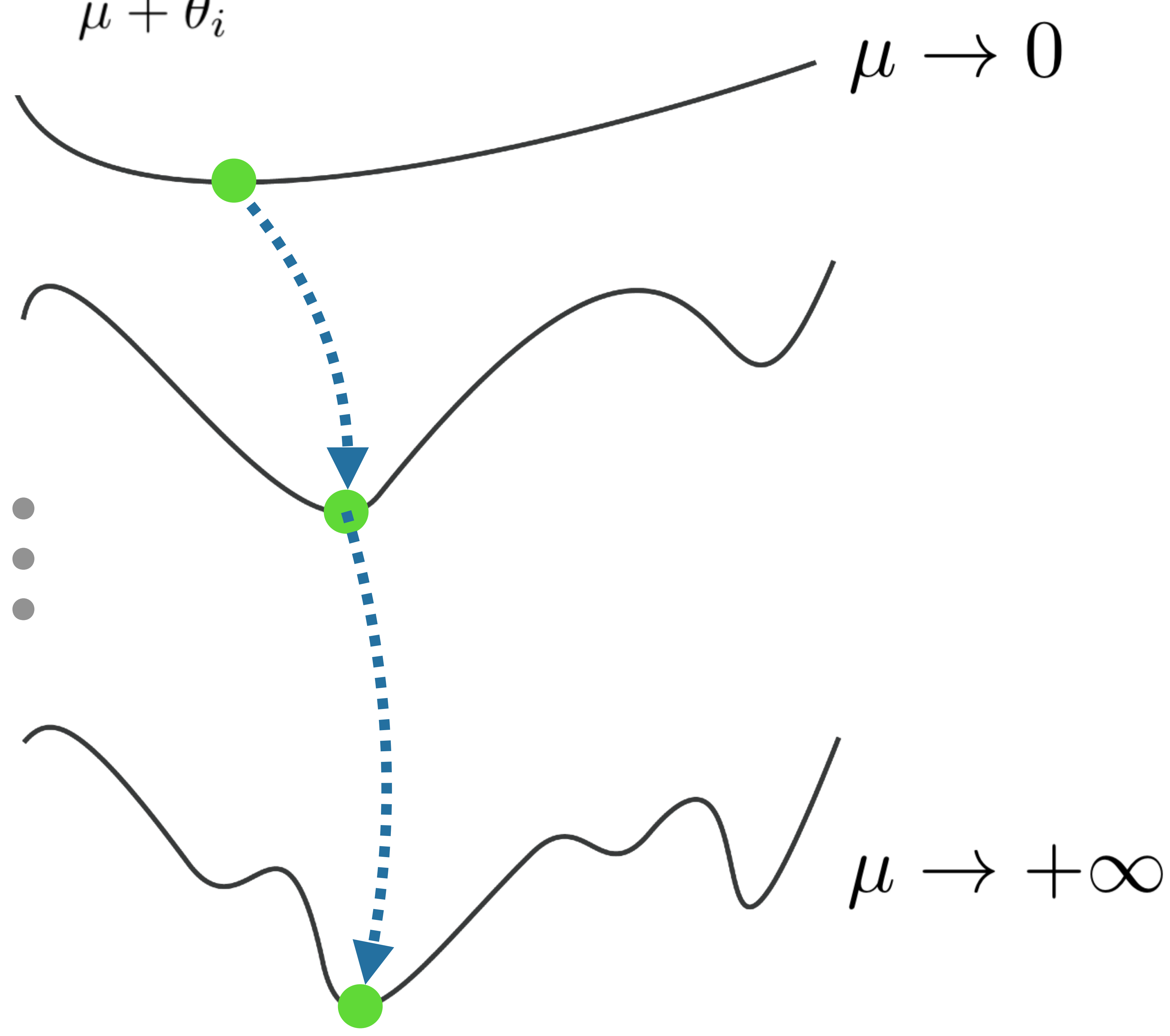


Graduated Non-Convexity (GNC)	Intuition

Define surrogate function with parameter μ

1. **Initialization:** set $\mu \rightarrow 0$
   - a. Set all weights $\theta_i = 1$
   - b. Variable Update (weighted least square)
2. While cost function decrease
   - a. Weight Update (closed-form)
   - b. Variable Update (weighted least square)
   - c. Increase Non-Convexity: $\mu_t = \delta \cdot \mu_{t-1}, \delta > 1$

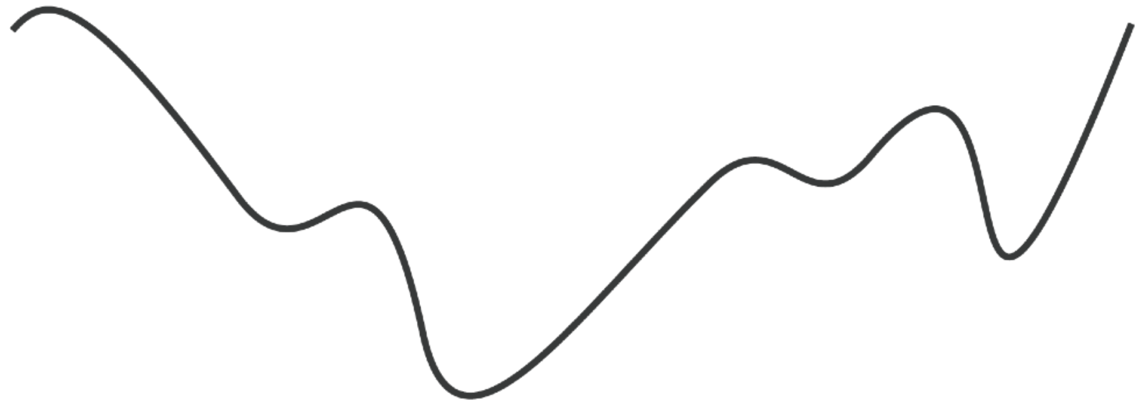
$$\arg\min_{\substack{x \in \mathbb{X}, \\ \theta_i \in [0,1], \forall i}} \sum_{i \in \mathcal{M}} \theta_i \| r_i(x, y_i) \|^2 + \frac{\mu(1 - \theta_i)}{\mu + \theta_i}\bar{c}^2$$



[Yang, Antonante, Tzoumas, Carlone. Graduated non-convexity for robust spatial perception: from non-minimal solvers to global outlier rejection. RAL 2020. (best paper in robot vision at ICRA 2020)]

Graduated Non-Convexity (GNC): Algorithm

$$\underset{\substack{x \in \mathbb{X}, \\ \theta_i \in \{0,1\}, \forall i}}{\arg\min} \sum_{i \in \mathcal{M}} \theta_i \| r_i(x, y_i) \|^2 + (1 - \theta_i) \bar{c}^2$$

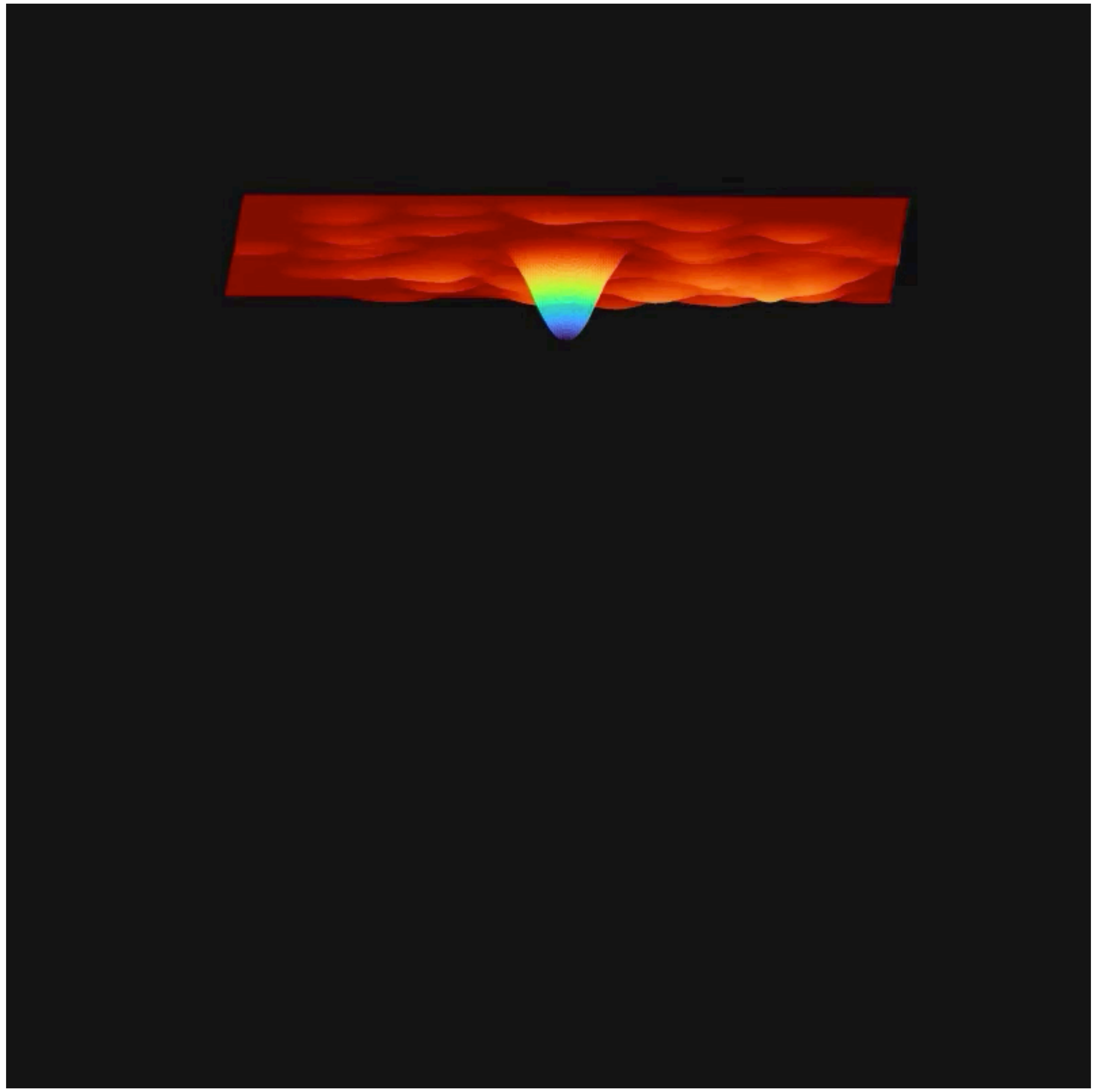


Graduated Non-Convexity (GNC)

Define surrogate function with parameter μ

1. **Initialization:** set $\mu \to 0$
   - a. Set all weights $\theta_i = 1$
   - b. Variable Update (weighted least square)
2. While cost function decrease
   - a. Weight Update (closed-form)
   - b. Variable Update (weighted least square)
   - c. Increase Non-Convexity: $\mu_t = \delta \cdot \mu_{t-1}, \delta > 1$

Intuition



[Yang, Antonante, Tzoumas, Carlone. Graduated non-convexity for robust spatial perception: from non-minimal solvers to global outlier rejection. RAL 2020. (best paper in robot vision at ICRA 2020)]

Shape#: Object localization in images using GNC

Truncated Least Squares

$$\arg\min_{\substack{x\in\mathbb{X},\\ \theta_i\in\{0,1\},\forall i}} \sum_{i\in\mathcal{M}} \theta_i \|r_i(x,y_i)\|^2 + (1-\theta_i)\bar{c}^2$$

Image-based Object Localization (shape alignment)

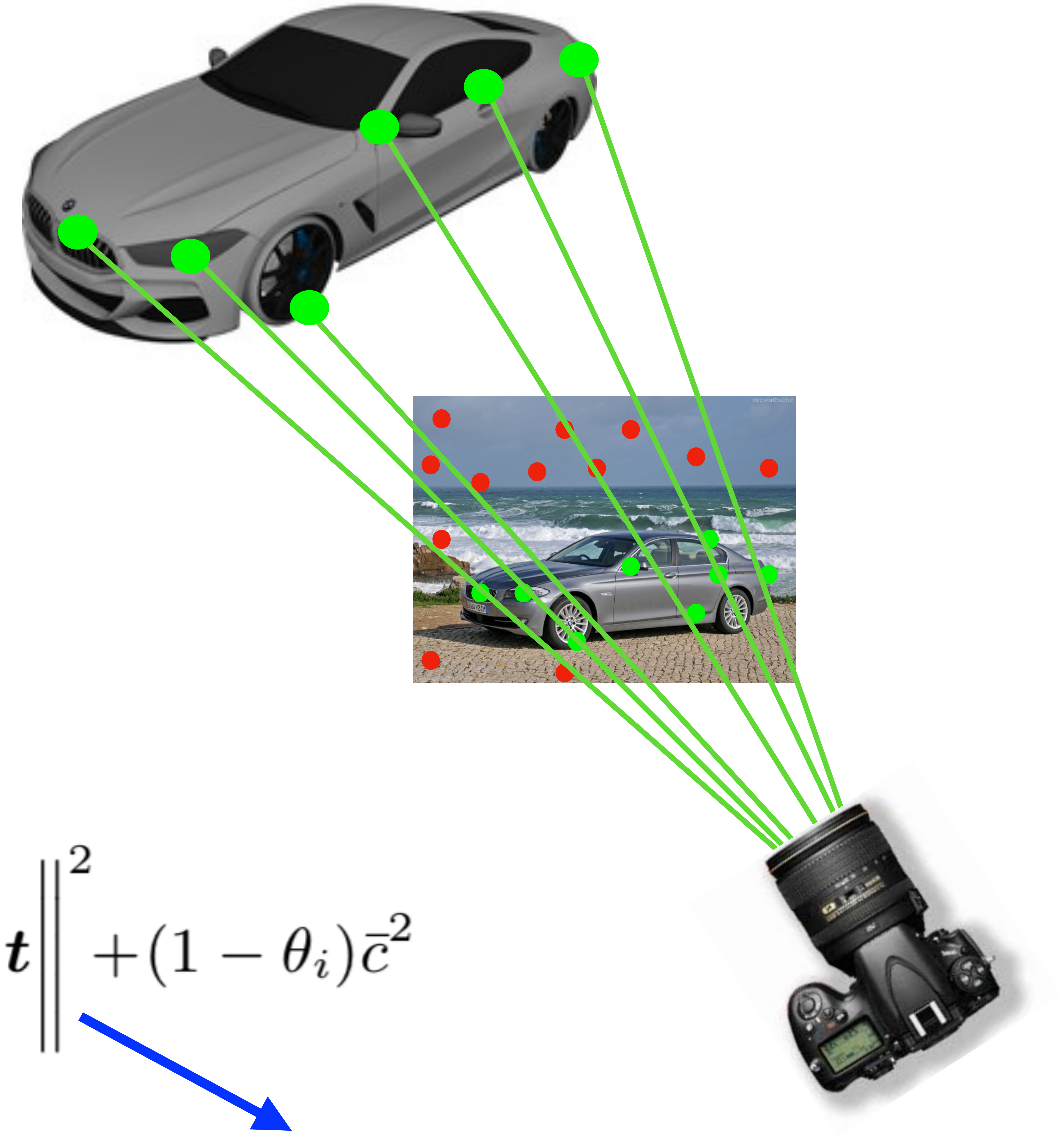
$$\min_{\substack{\beta_k\geq 0, k=1,\dots,K\\ \boldsymbol{t}\in\mathbb{R}^2, \boldsymbol{R}\in\mathrm{SO}(3)\\ \theta_i\{0,1\},\forall i}} \sum_{i=1}^{N} \theta_i \left\| \boldsymbol{z}_i - \Pi\boldsymbol{R}\left(\sum_{k=1}^{K}\beta_k \boldsymbol{B}_{ki}\right) - \boldsymbol{t}\right\|^2 + (1-\theta_i)\bar{c}^2$$

Pixel measurement

Projection

3D object template

(***R*, *t***): unknown object pose

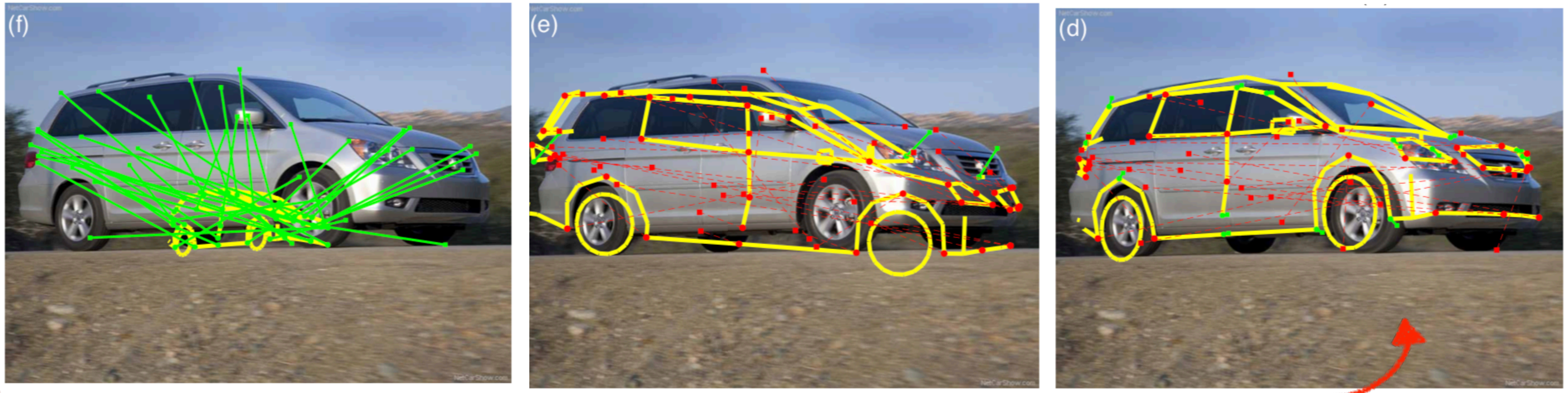


Shape#: Object localization in images using GNC

Shape alignment:

- FG3DCar dataset
- 300 car images with corresponding CAD models

[Zhou, CVPR’15] RANSAC Shape#



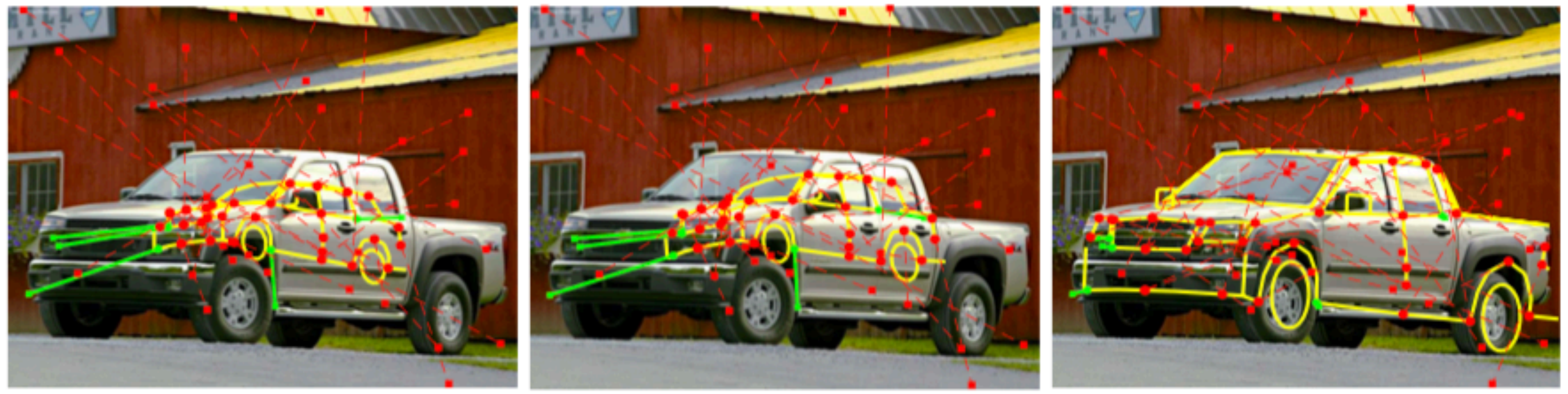
70% outliers

Proposed

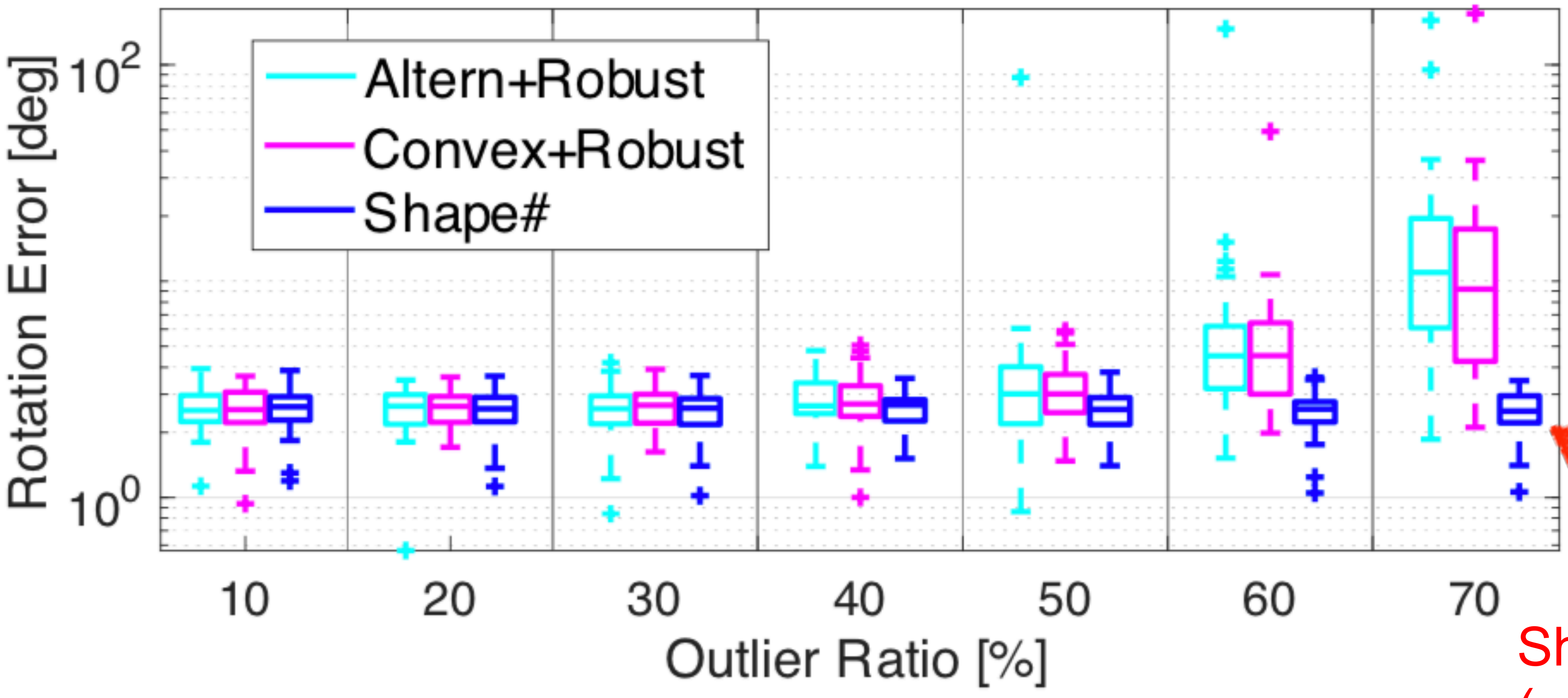
Proposed

Altern+Robust Convex+Robust Shape#

(a) Chevrolet Colorado LS 40% outliers.



(b) Chevrolet Colorado LS 70% outliers.



Shape# (proposed) is best approach

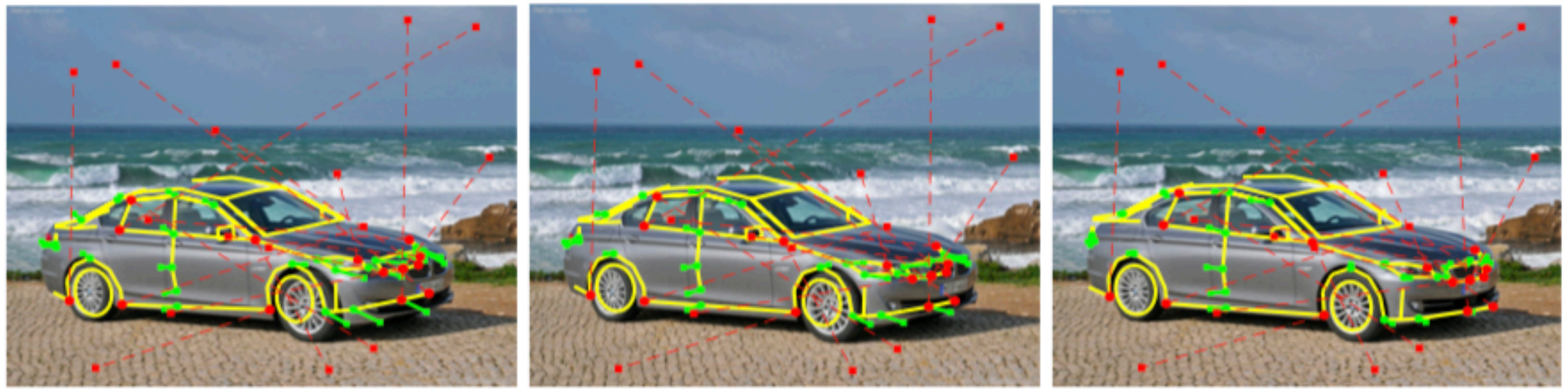
<- Estimate pose and shape of car

[Yang and Carlone. In Perfect Shape: Certifiably Optimal 3D Shape Reconstruction from 2D Landmarks. CVPR 2020]

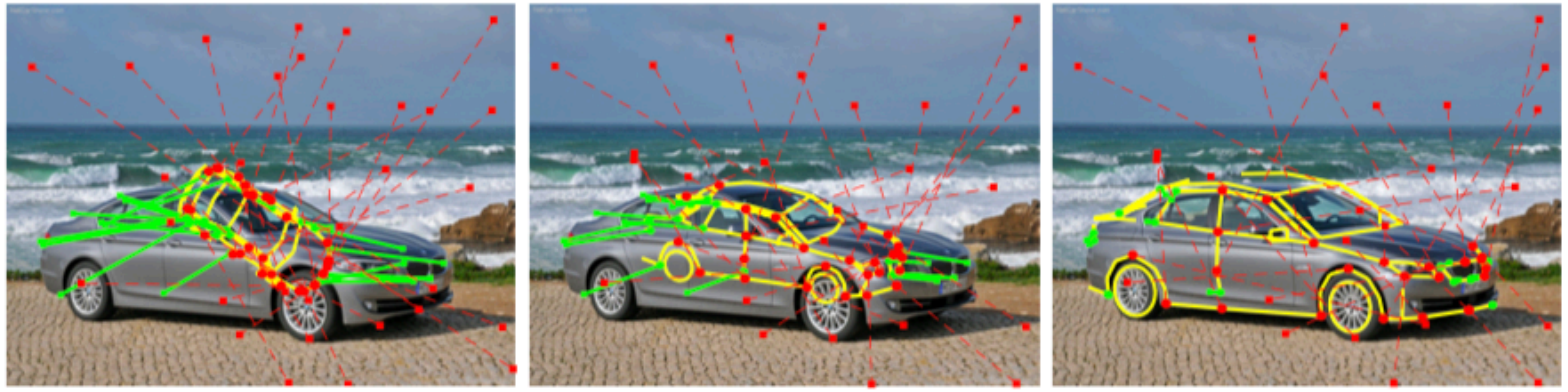
Shape#: experimental results

Proposed

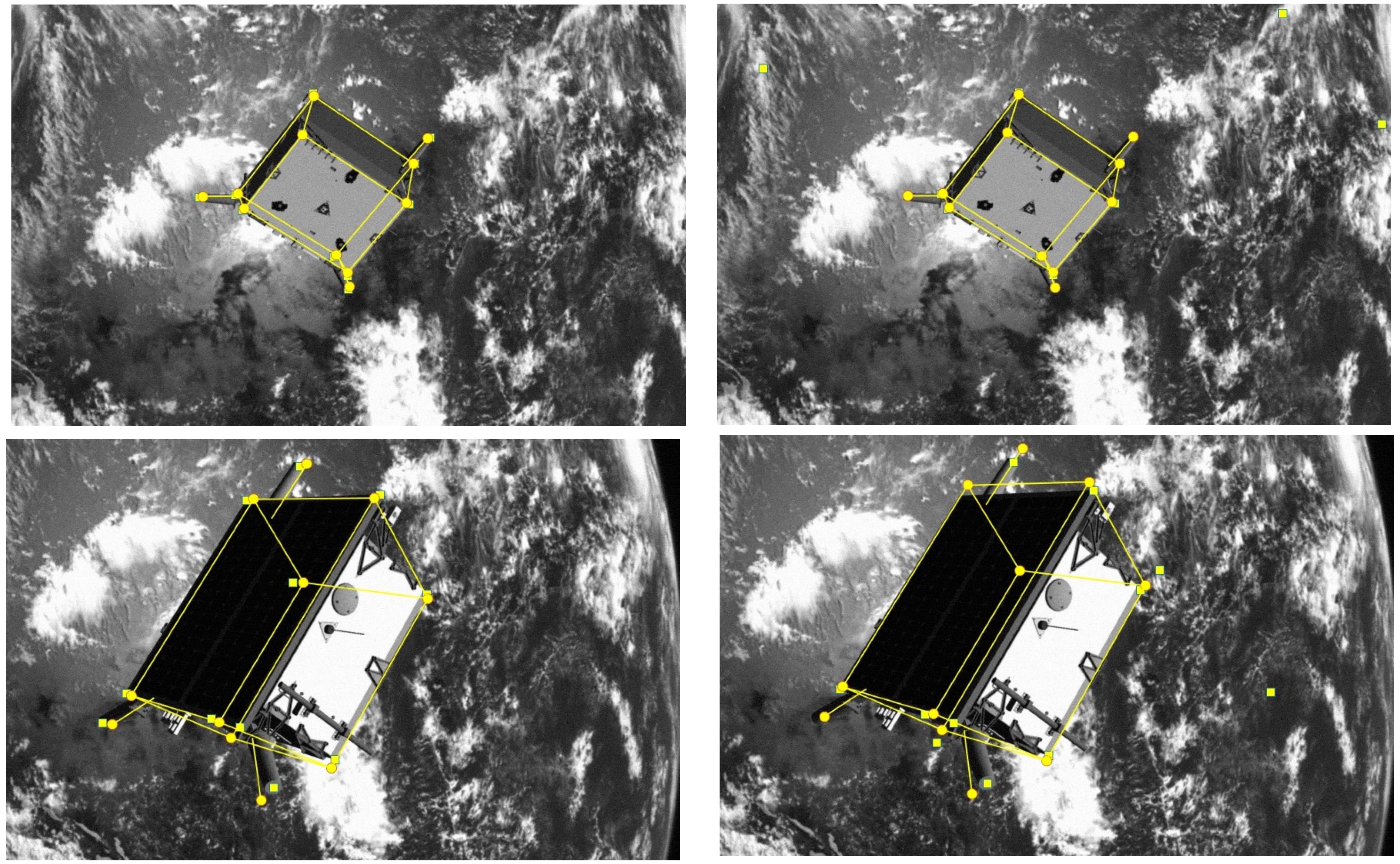
Altern+Robust Convex+Robust Shape#



(c) BMW 5-Series 40% outliers.



(d) BMW 5-Series 70% outliers.



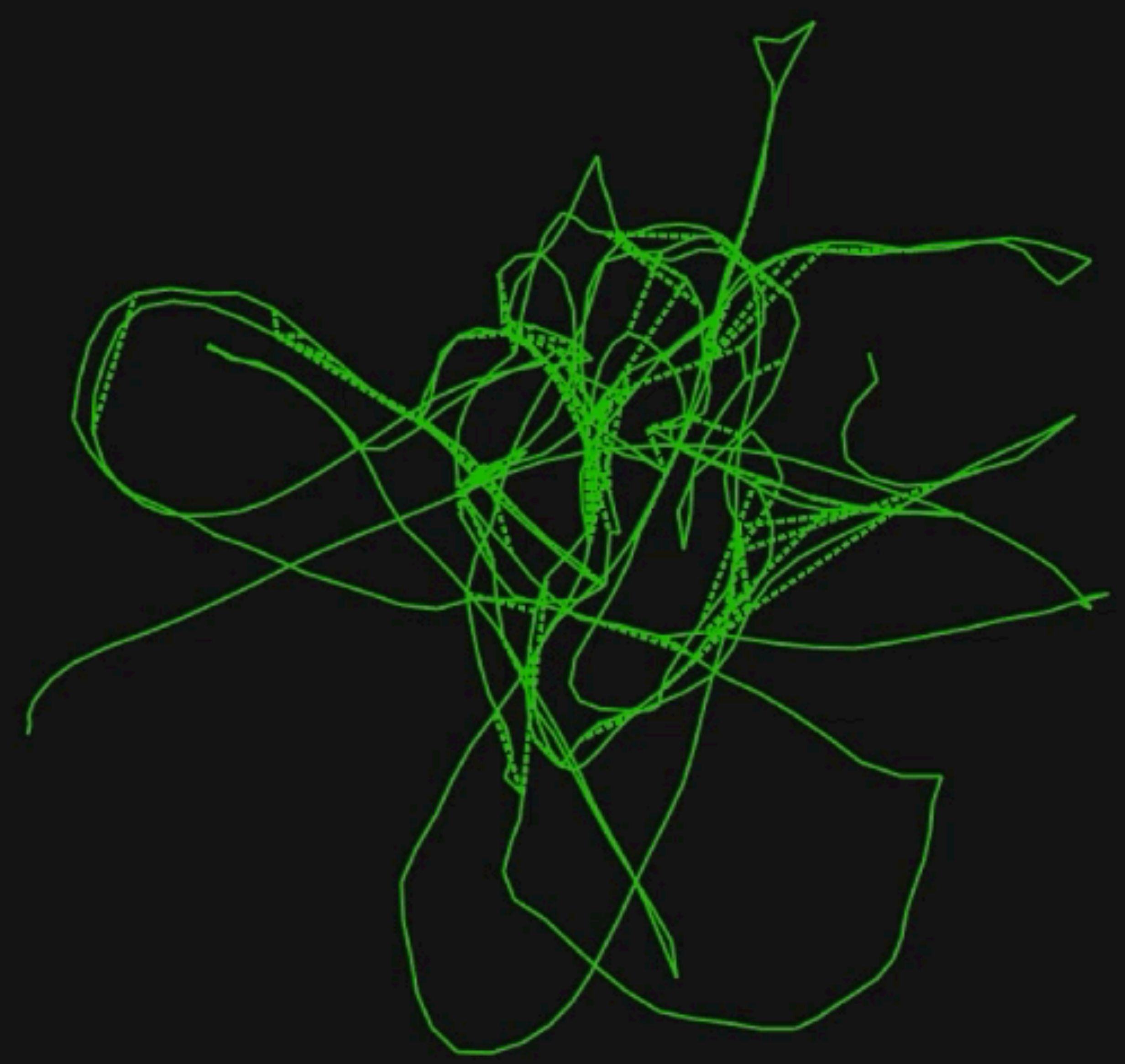
- Certifiable spacecraft pose estimation
- SPEED benchmarking dataset

[Yang and Carlone. In Perfect Shape: Certifiably Optimal 3D Shape Reconstruction from 2D Landmarks. CVPR 2020]
[Yang and Carlone, One Ring to Rule Them All: Certifiably Robust Geometric Perception with Outliers, NeurIPS'20.]

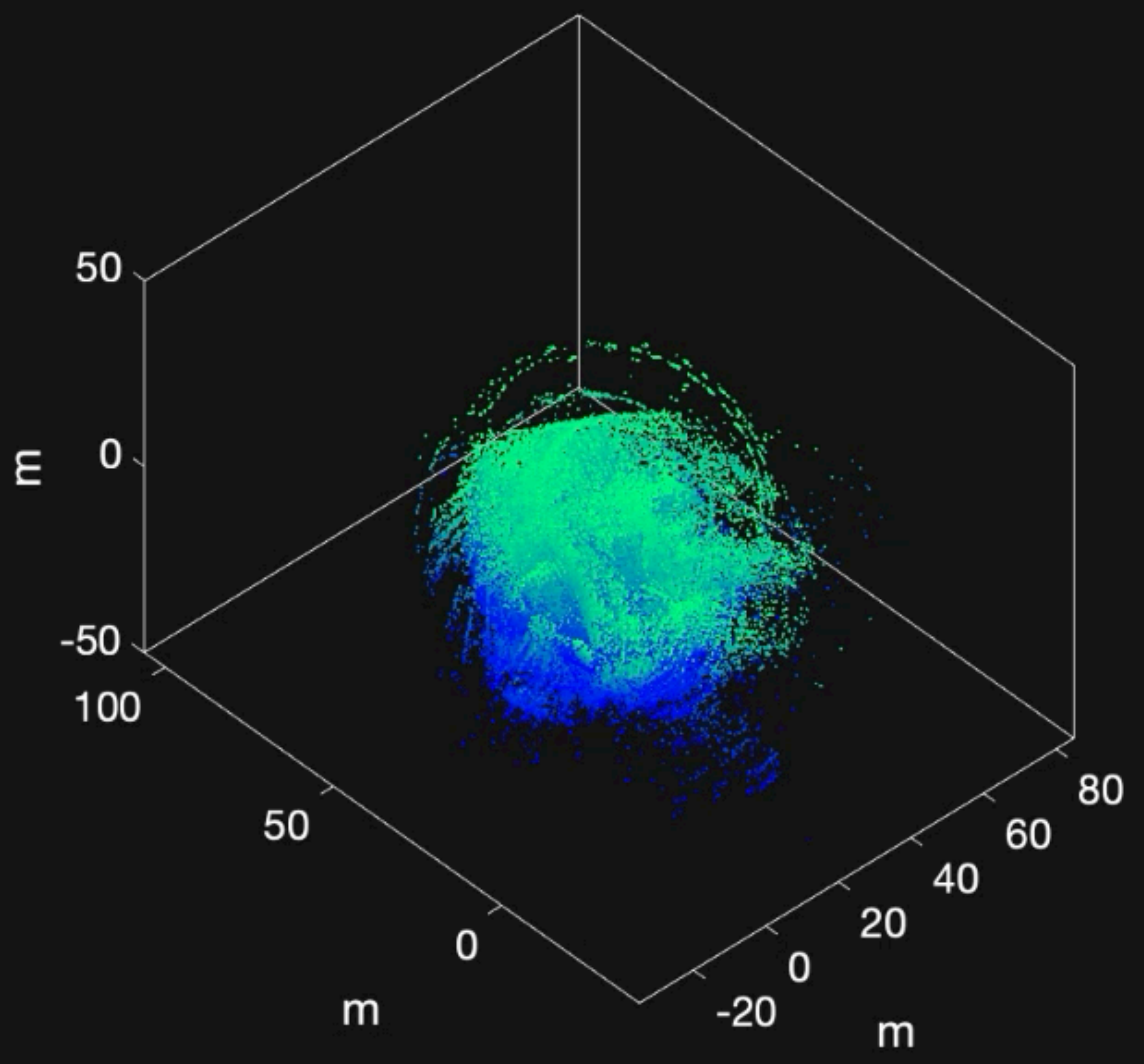
GNC for Simultaneous Localization and Mapping

Robot Trajectory

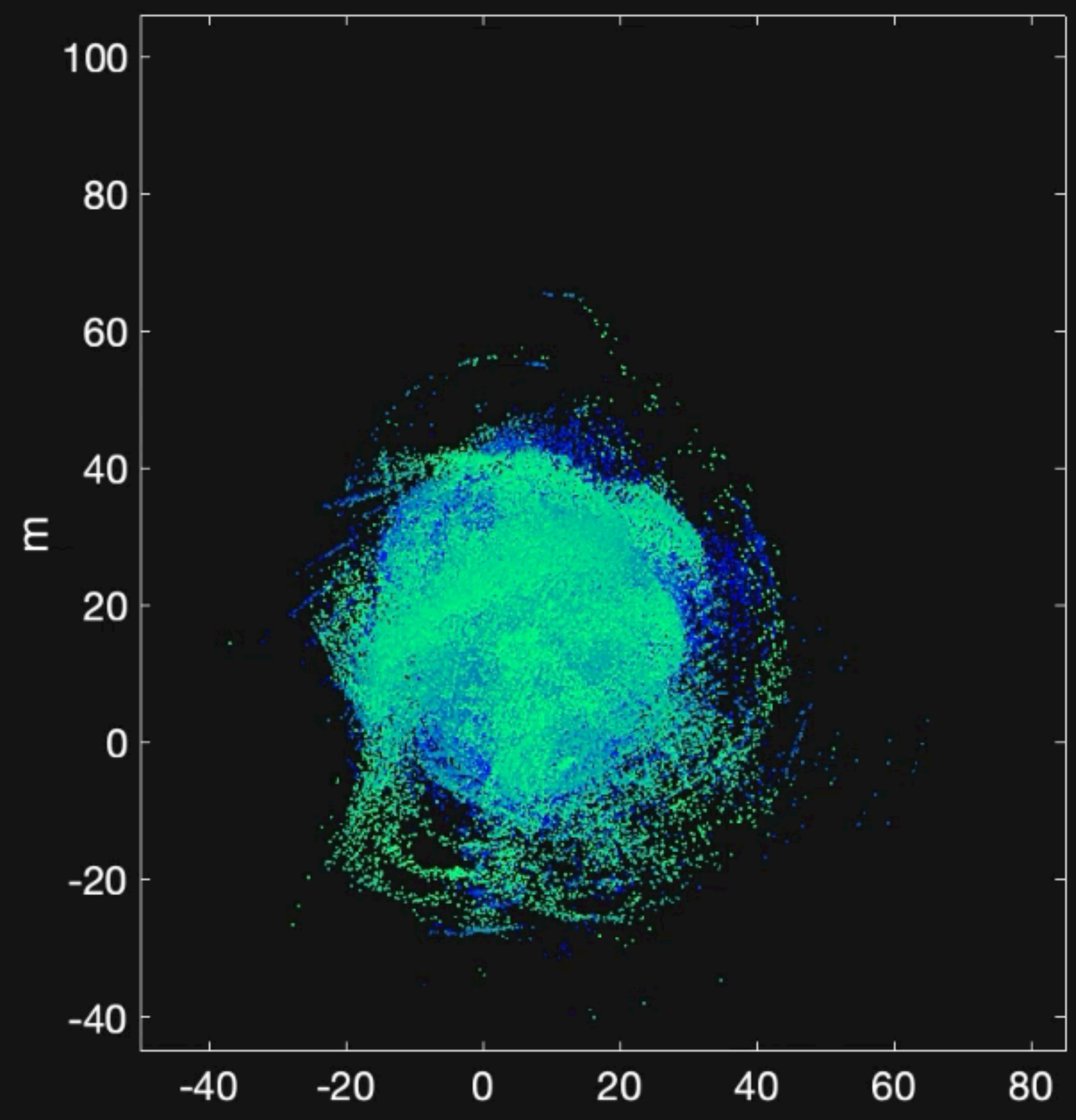
(Outliers in red)



Reconstructed Map



Top View



Problem: estimate trajectory given motion estimates and loop closures.
Loop closures are contaminated with outliers

Inputs

SPARK Lab

[Yang, Antonante, Tzoumas, Carlone. Graduated non-convexity for robust spatial perception: from non-minimal solvers to global outlier rejection. RAL 2020. (best paper in robot vision at ICRA 2020)]

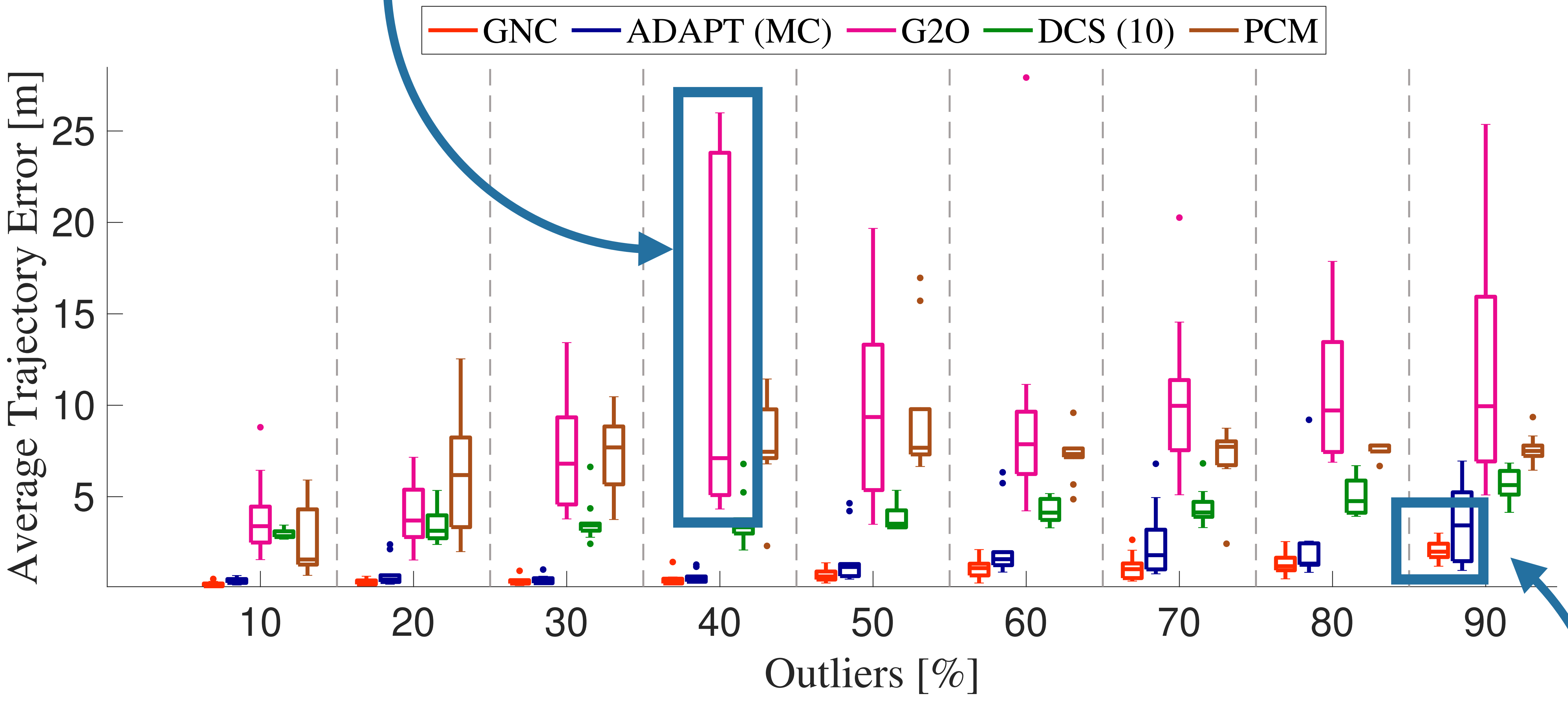
GNC for Simultaneous Localization and Mapping

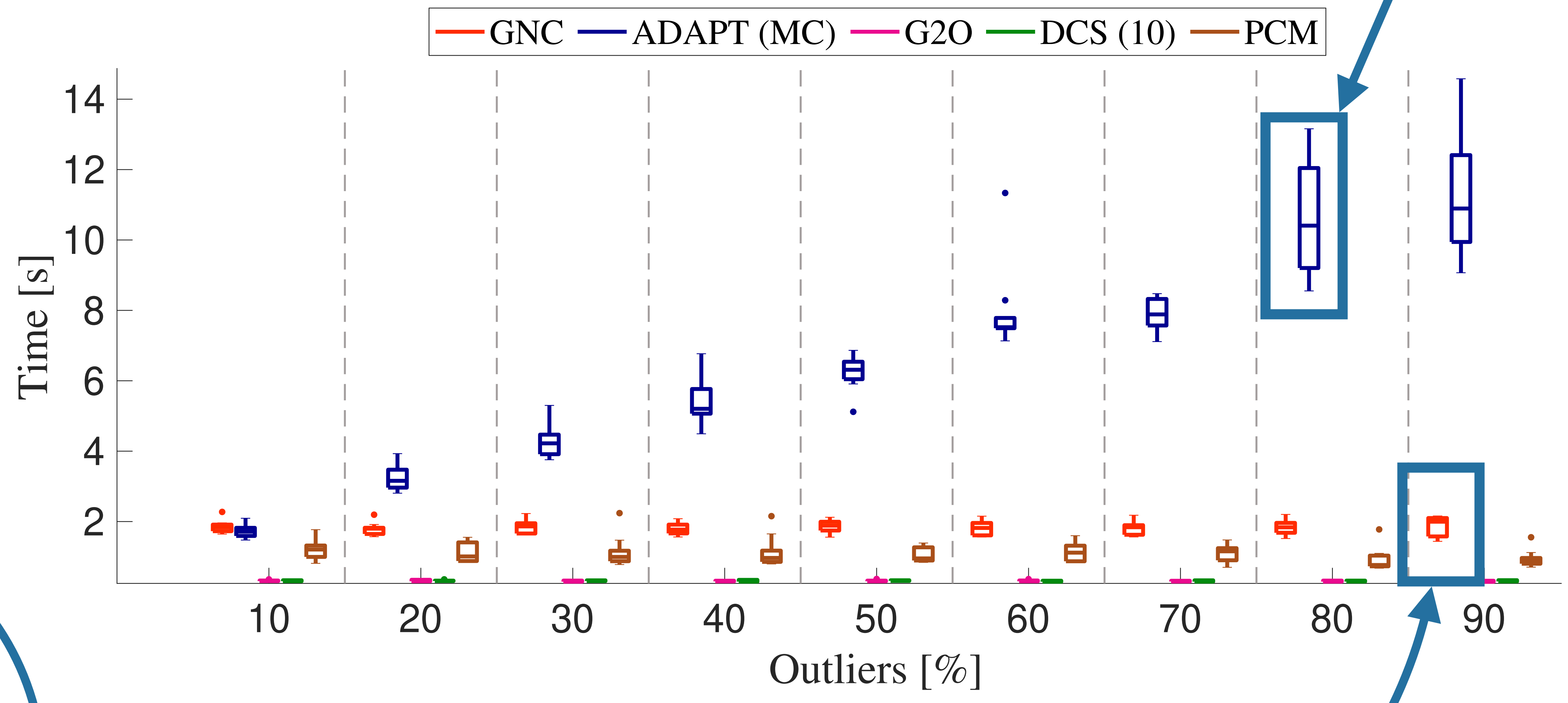
Accuracy

Runtime

G2O: non-robust solver

baseline grows linearly





CSAIL SLAM dataset

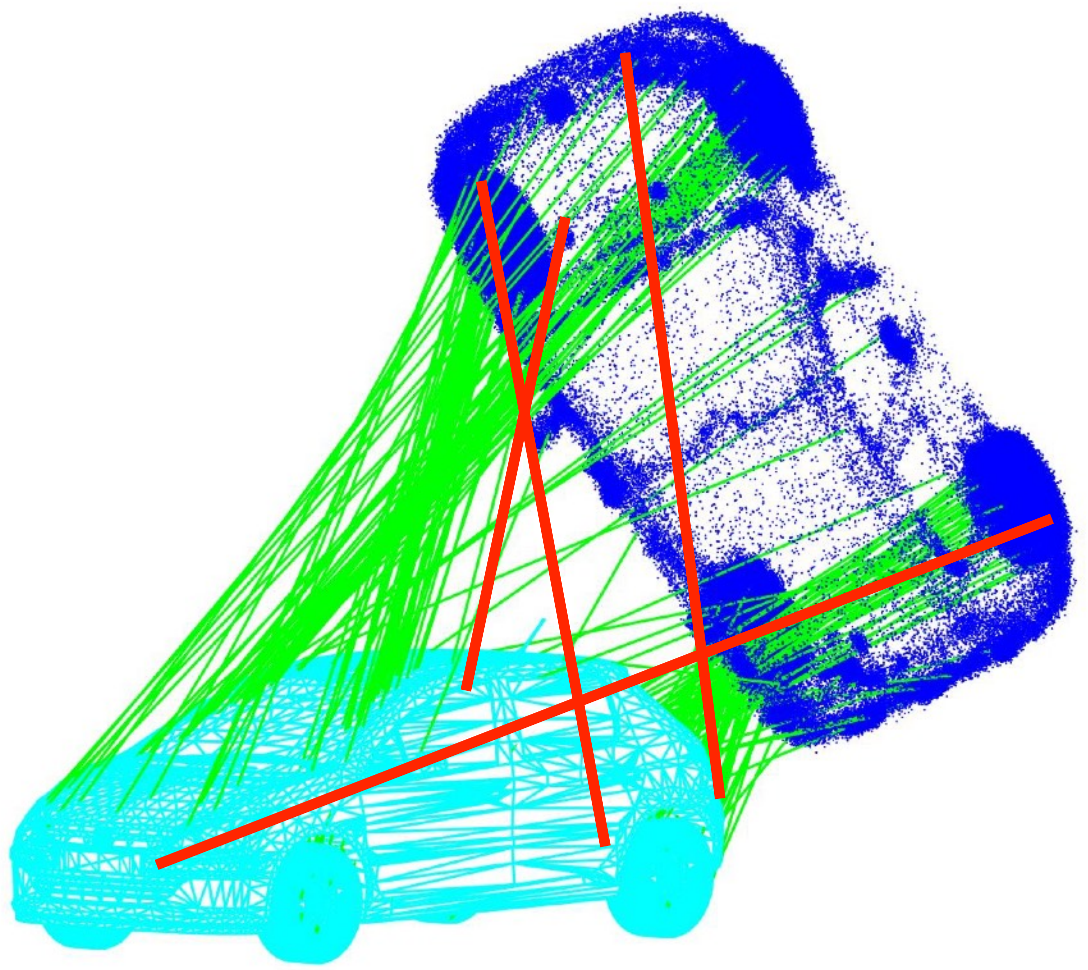
GNC robust
up to 90% outliers

GNC constant

[Yang, Antonante, Tzoumas, Carlone. Graduated non-convexity for robust spatial perception: from non-minimal solvers to global outlier rejection. RAL 2020. (best paper in robot vision at ICRA 2020)]
[Antonante, Tzoumas, Yang, Carlone. Outlier-Robust Estimation: Hardness, Minimally-Tuned Algorithms, and Applications. arXiv 2021]

GNC: summary

Mesh Registration

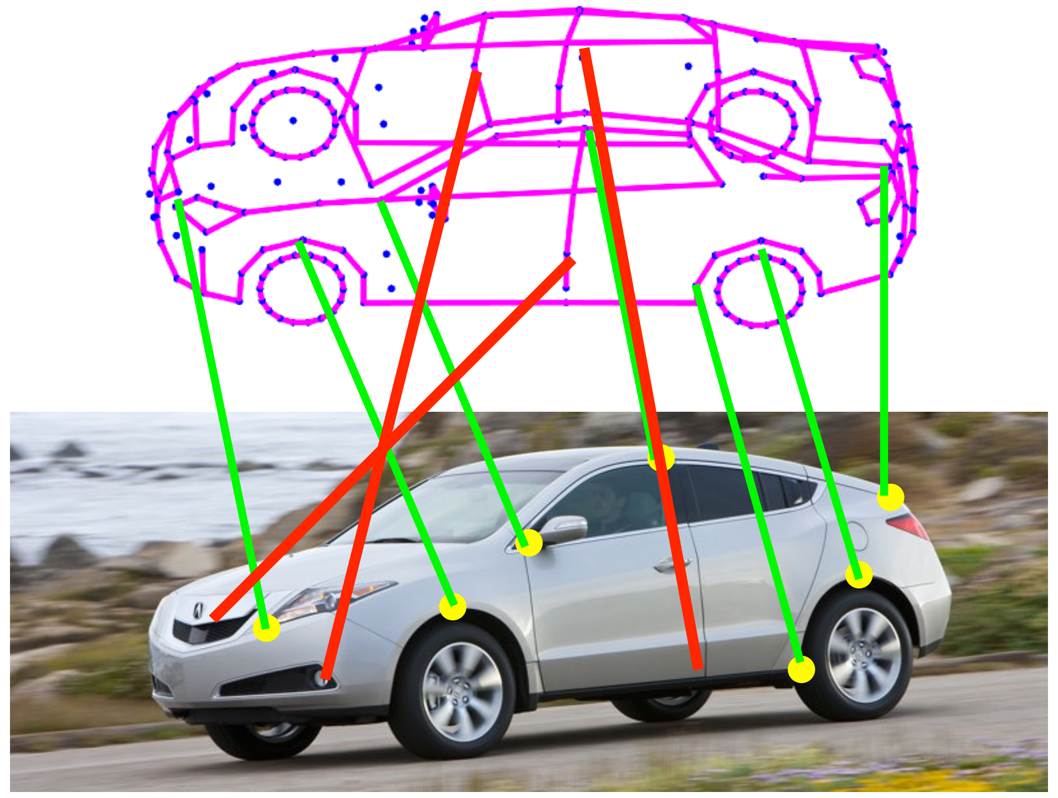


up to

80%

outliers

Shape Alignment

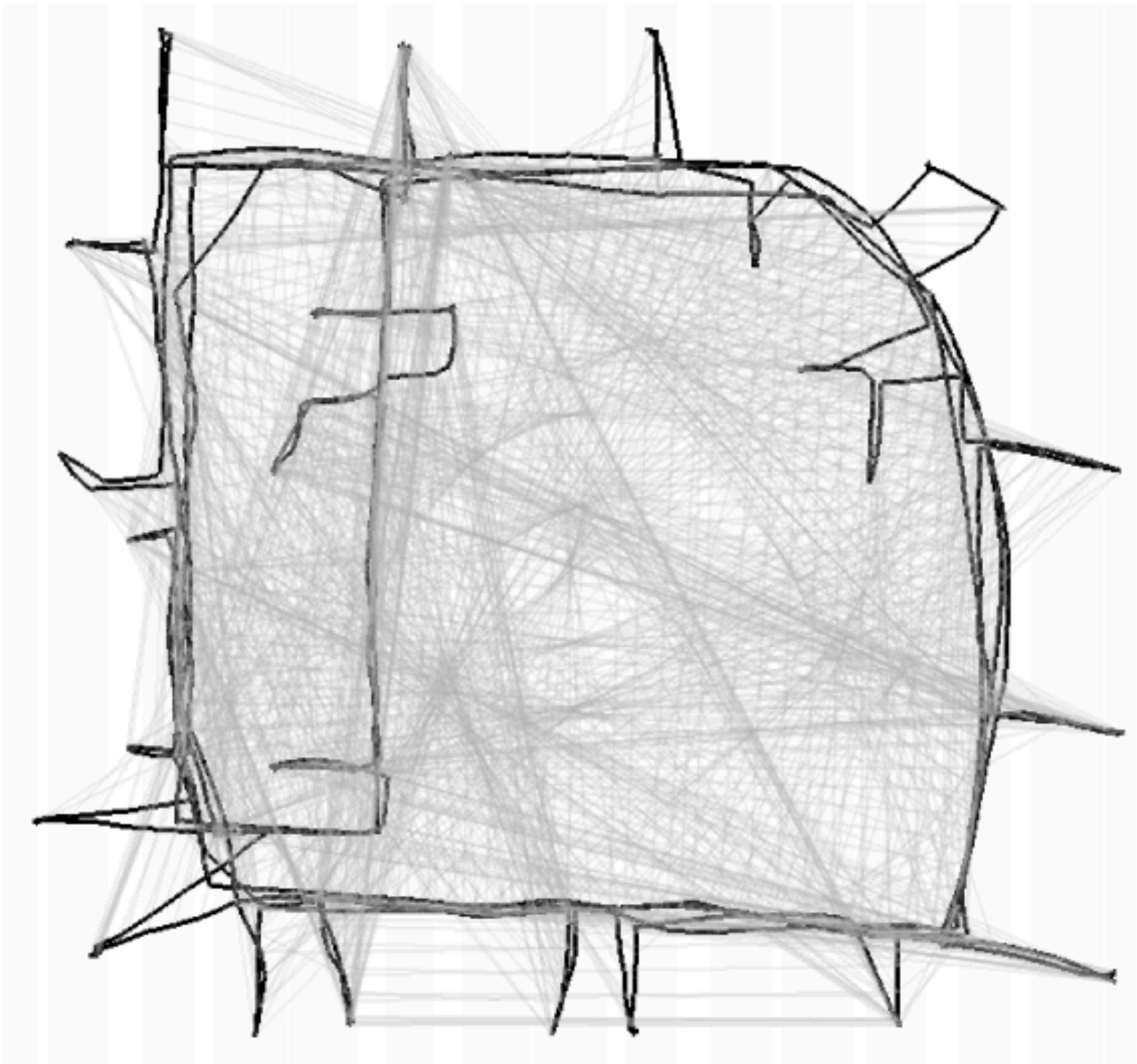


up to

70%

outliers

SLAM



up to

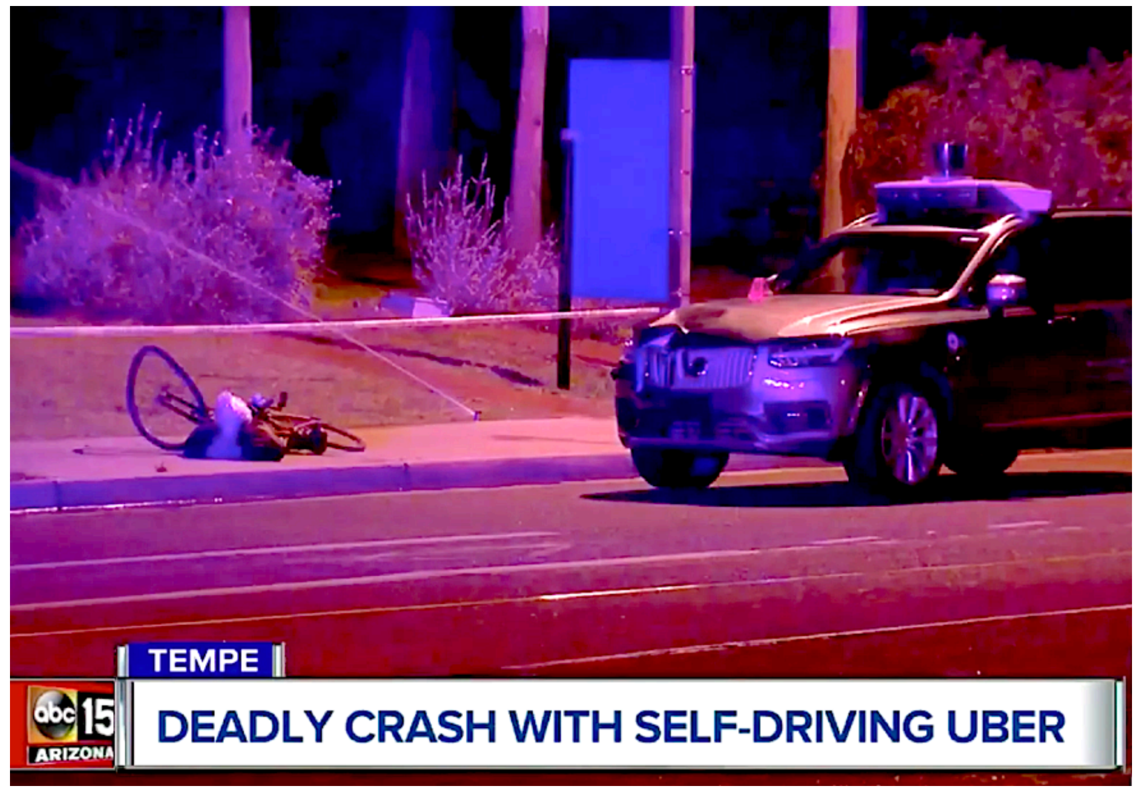
90%

outliers

No need for initial guess (as opposed to local solvers)
No need for minimal solver (as opposed to RANSAC)
GNC implementation available in Matlab and GTSAM

[Yang, Antonante, Tzoumas, Carlone. Graduated non-convexity for robust spatial perception: from non-minimal solvers to global outlier rejection. RAL 2020. (best paper in robot vision at ICRA 2020)]

Outline



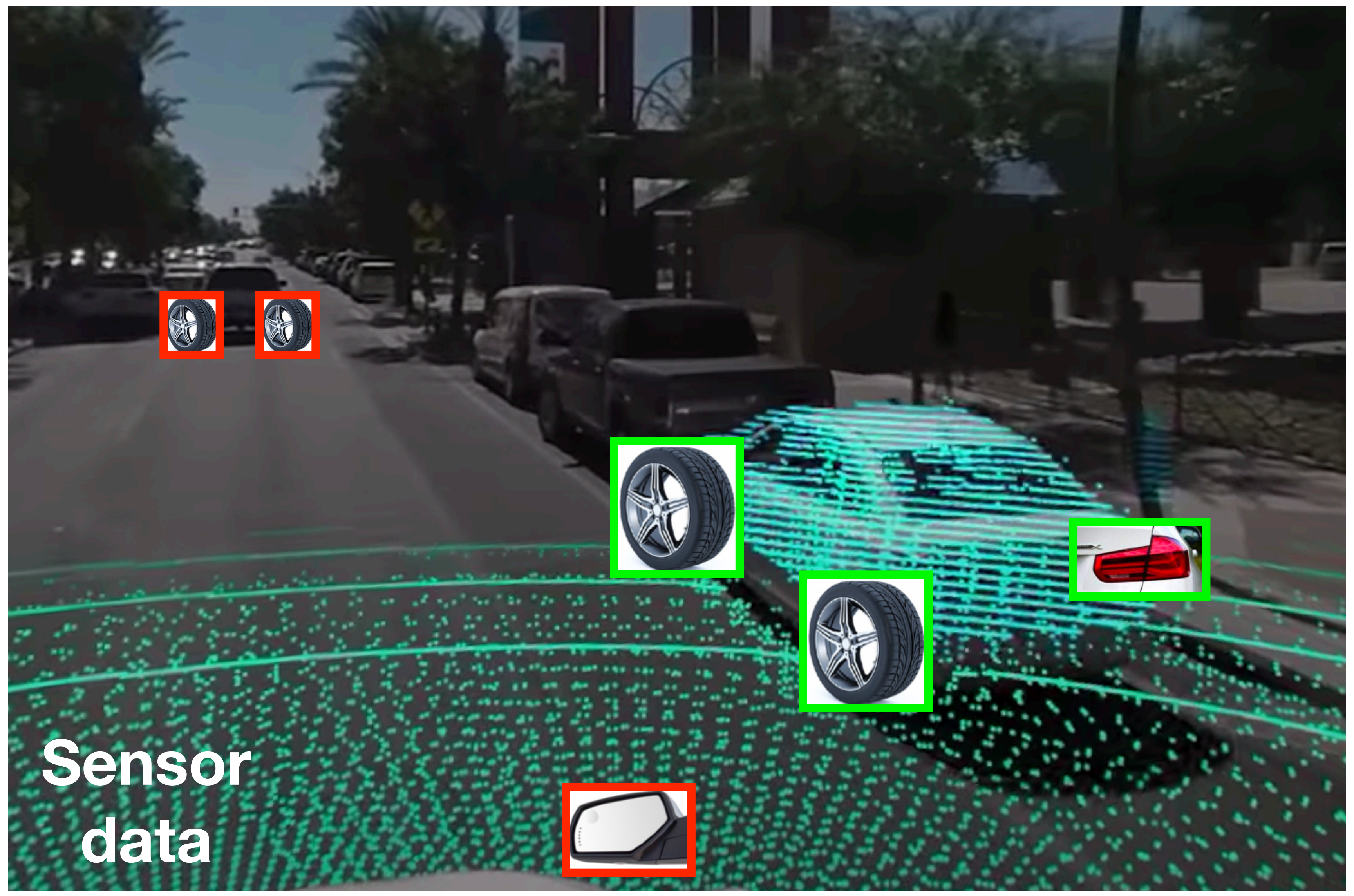
What is a "certifiable algorithm" for geometric perception?

- Graduated Non-Convexity
  [ICRA'20, CVPR'20]
- Graph-theoretic outlier pruning
  [TRO'20, ICRA'21, arxiv'21]
- Optimality certification
  [TRO'20, NeurIPS'20]

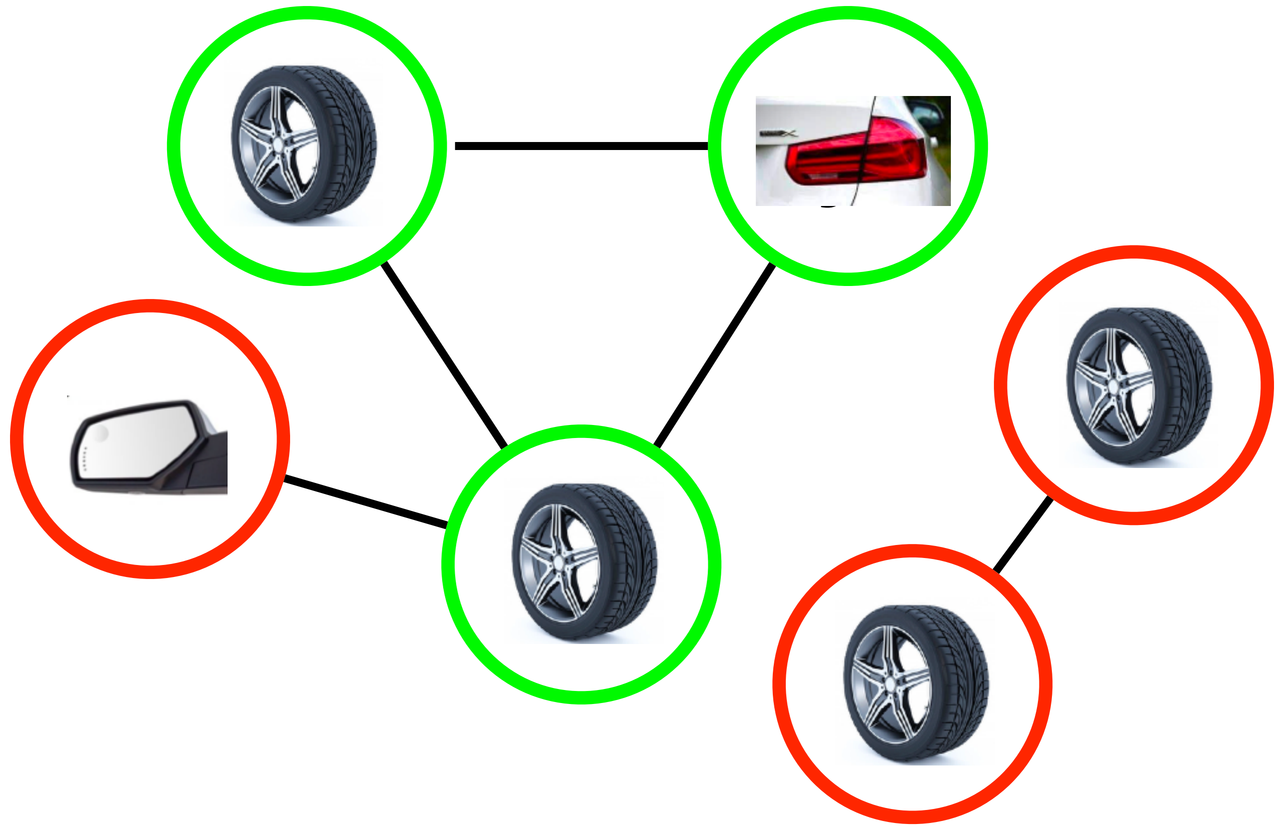
Can we further boost the robustness of GNC?

Insight: we can prune measurements that are mutually incompatible

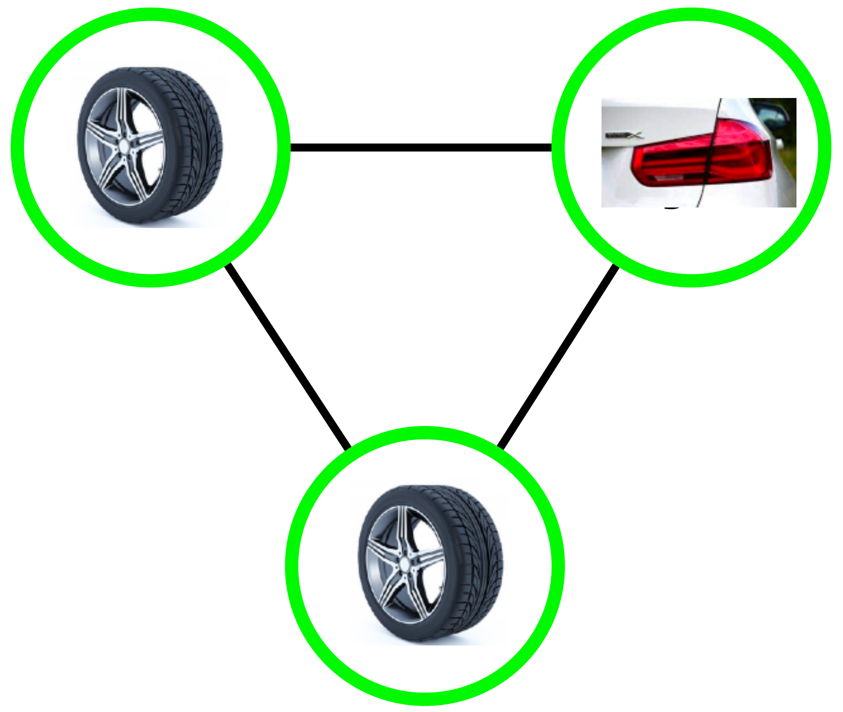
Object
To detect



Compatibility Graph



Maximum Clique



ROBIN: Reject Outliers Based on INvariants

- Used as pre-processing for GNC, boosts robustness to 90-95% outliers
- Works beyond pairwise comparisons
- Fast implementation (sparse graphs)
- Does not discard inliers if we use proper invariants

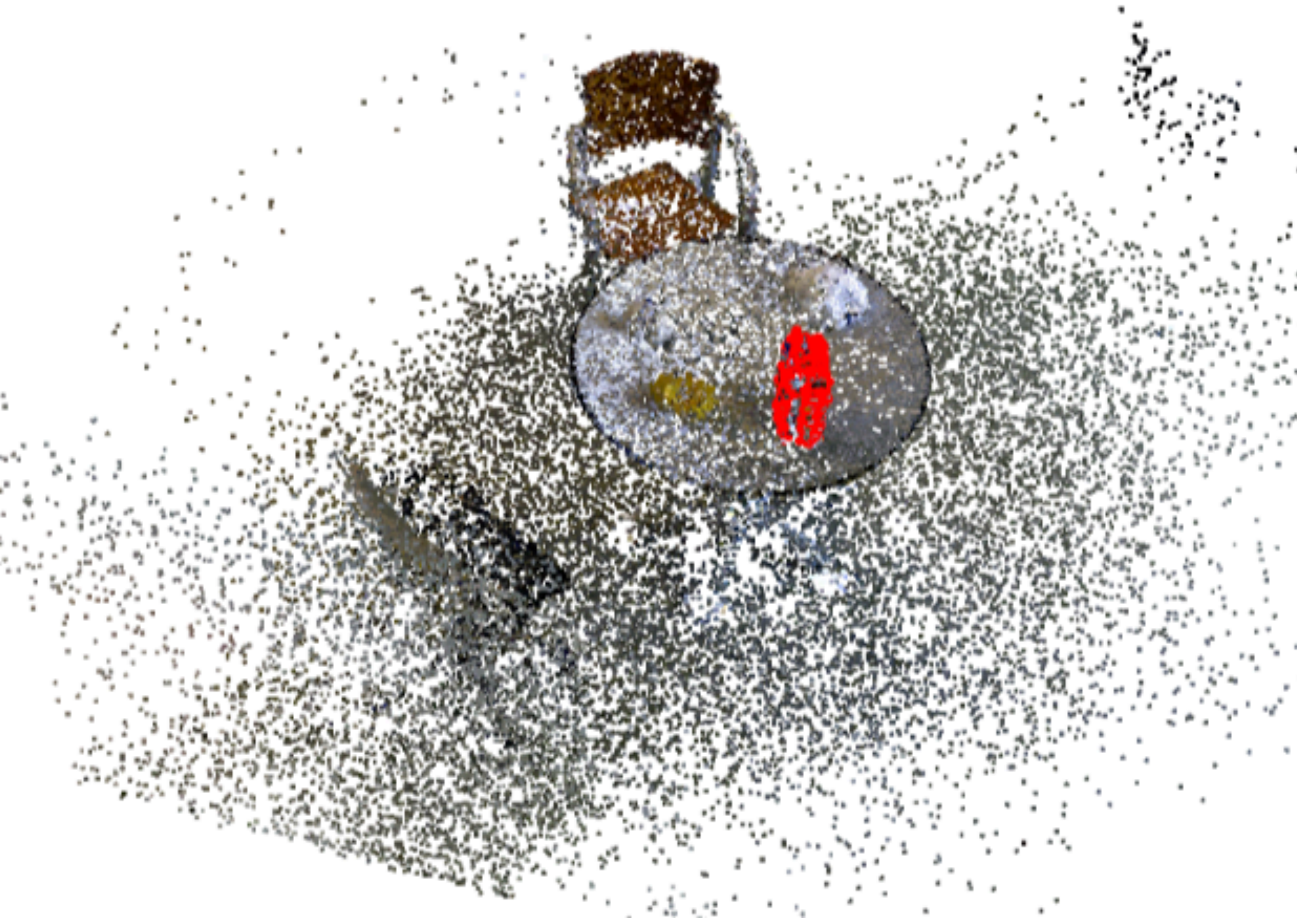
[Shi, Yang, Carlone. ROBIN: a Graph-Theoretic Approach to Reject Outliers in Robust Estimation using Invariants. ICRA 2021]

TEASER++: object localization in point clouds

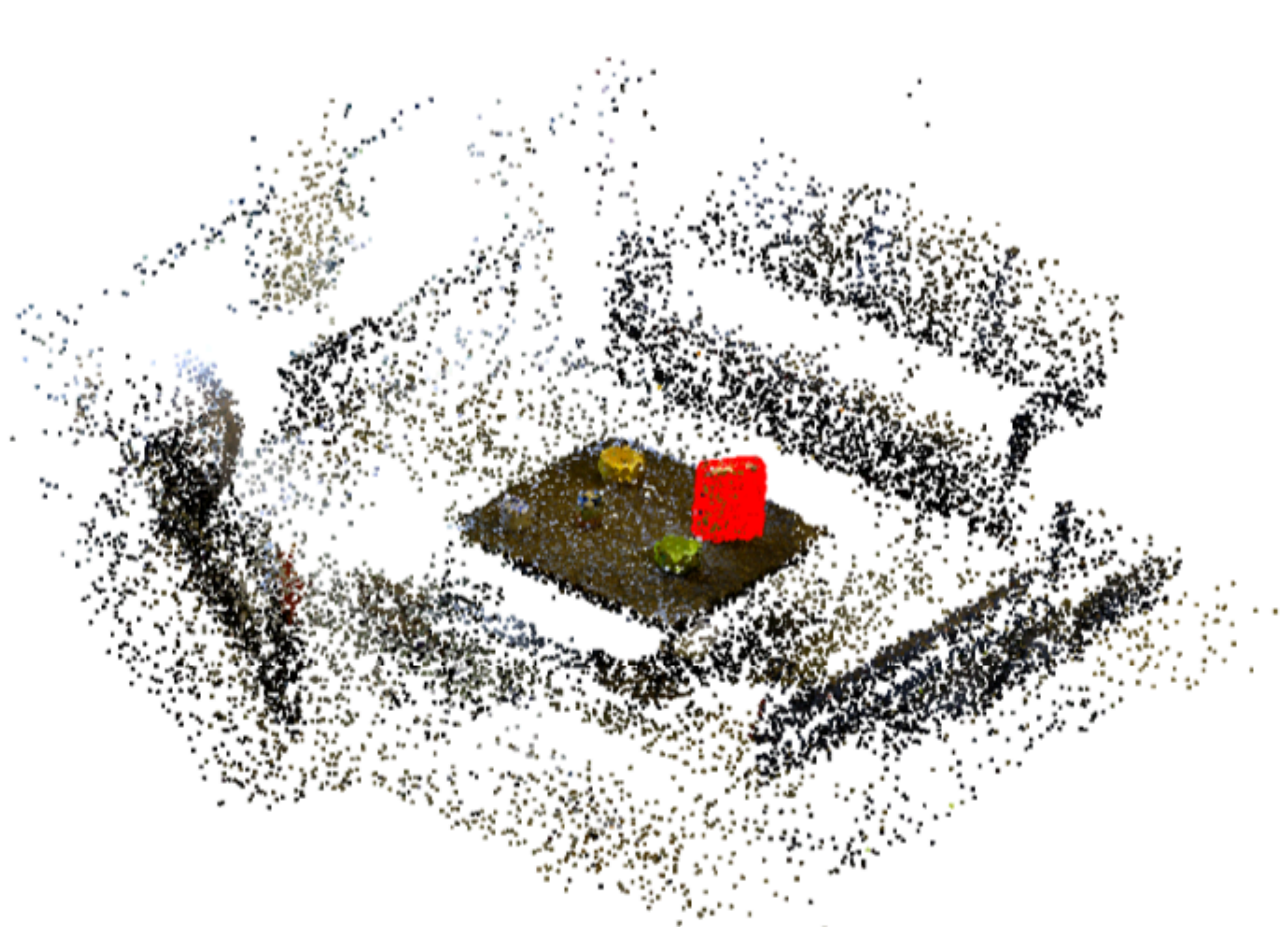
Registration with ROBIN + GNC

- RGB-D point cloud dataset
- ~500 FPFH features

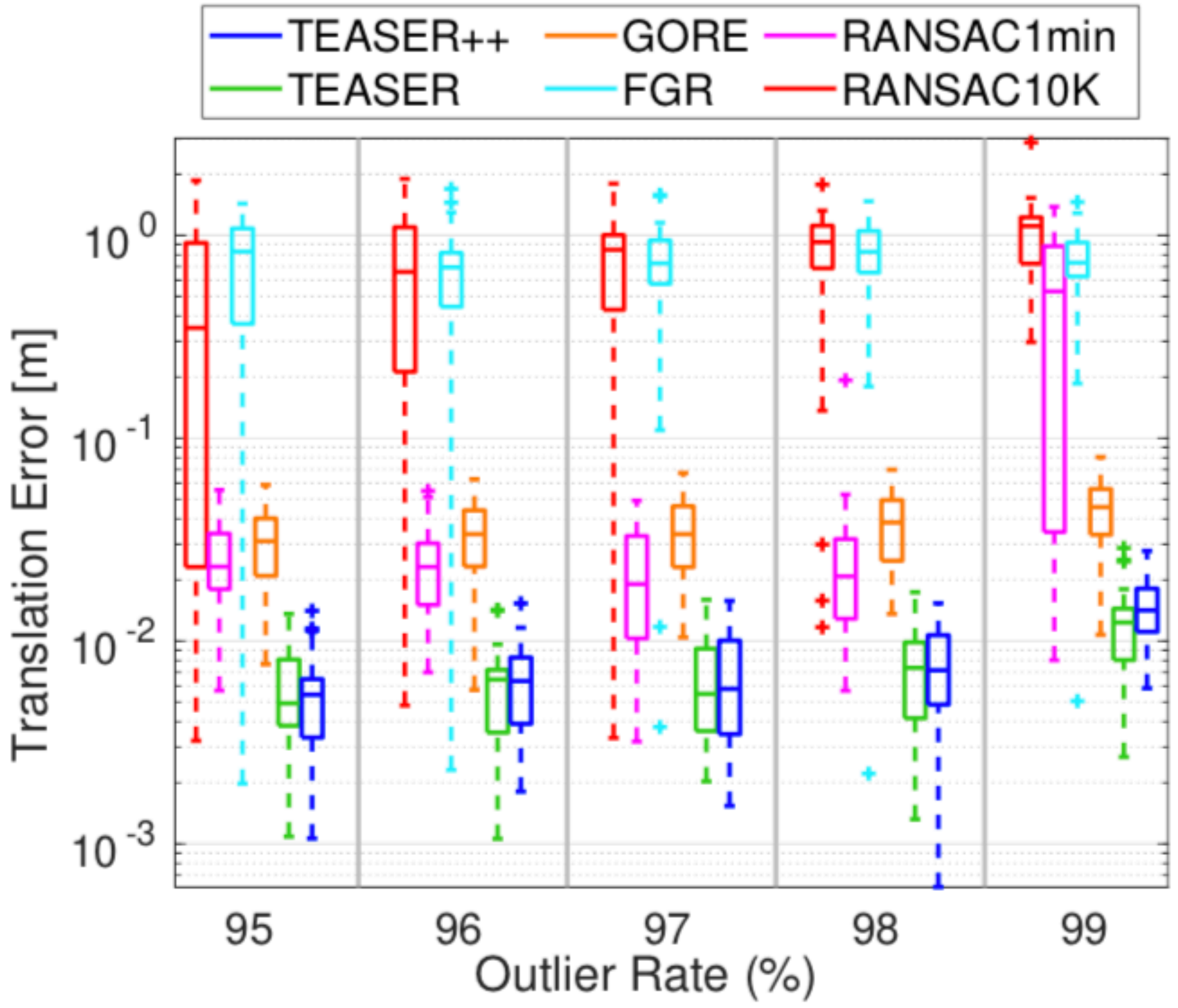
96.87% outliers



97.37% outliers



95.44% outliers



TEASER and TEASER++ (proposed) are best approaches

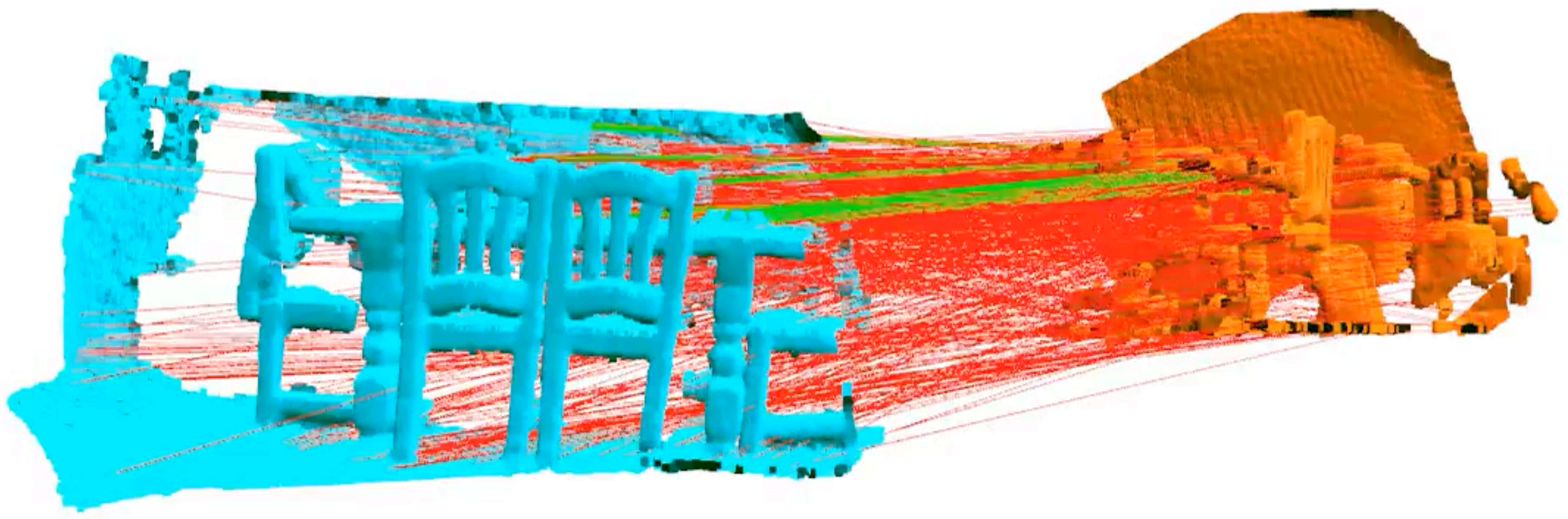
TEASER++ runs in 20ms

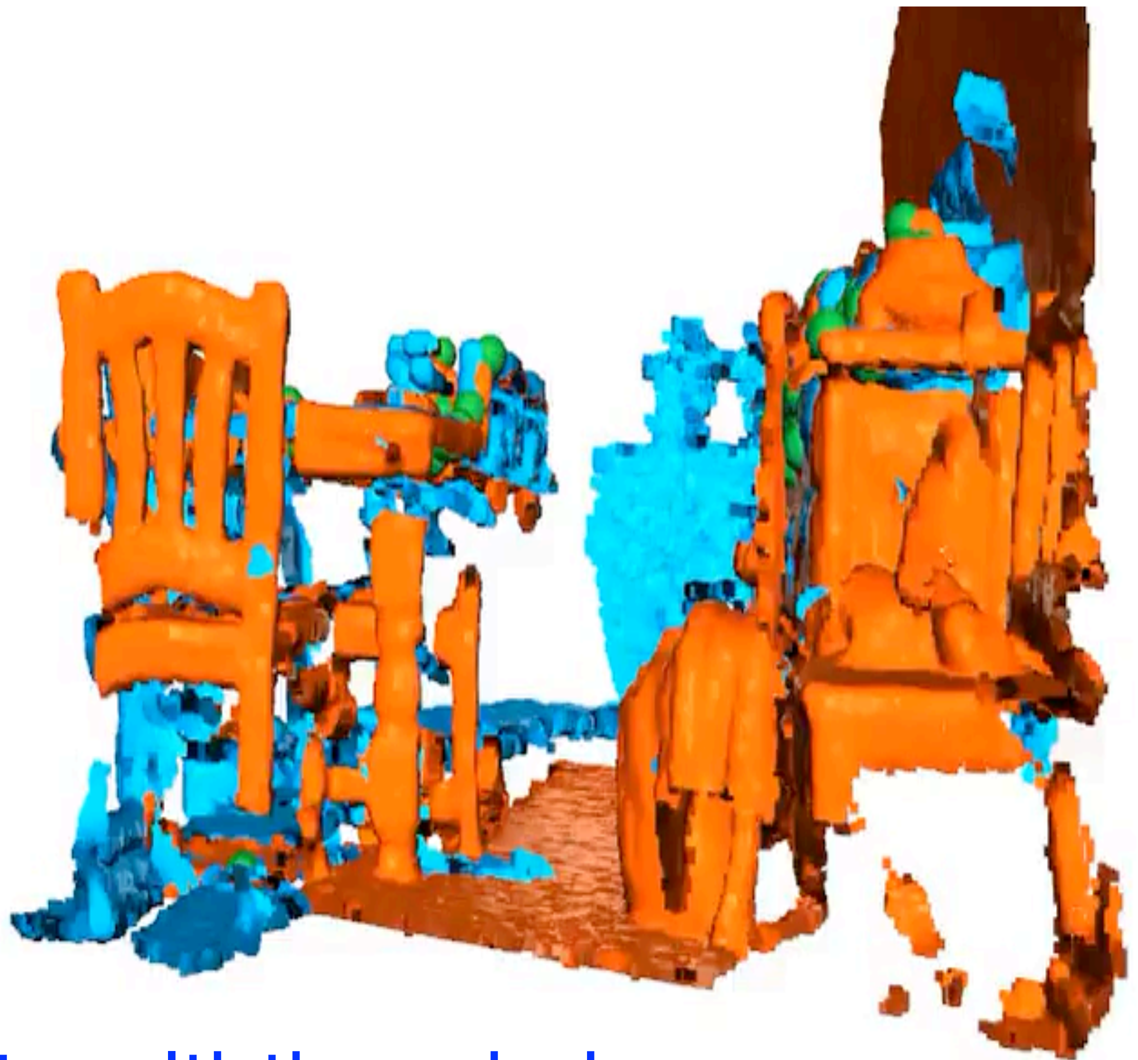
TEASER++: aligning point clouds

- 3DMatch dataset, ~1000 deep-learned features (using 3DSmoothNet)

Success rate

Scenes	Kitchen (%)	Home 1 (%)	Home 2 (%)	Hotel 1 (%)	Hotel 2 (%)	Hotel 3 (%)	Study Room (%)	MIT Lab (%)
RANSAC-1K	90.9	91.0	73.1	88.1	80.8	87.0	79.1	81.8
RANSAC-10K	96.4	92.3	73.1	92.0	84.6	90.7	82.2	81.8
TEASER++	97.7	92.3	82.7	96.9	88.5	94.4	88.7	84.4
TEASER++ (CERT)	99.2	97.5	90.0	98.8	94.9	97.7	94.8	93.9





Green lines:
inlier correspondences

Red lines:
outlier correspondences

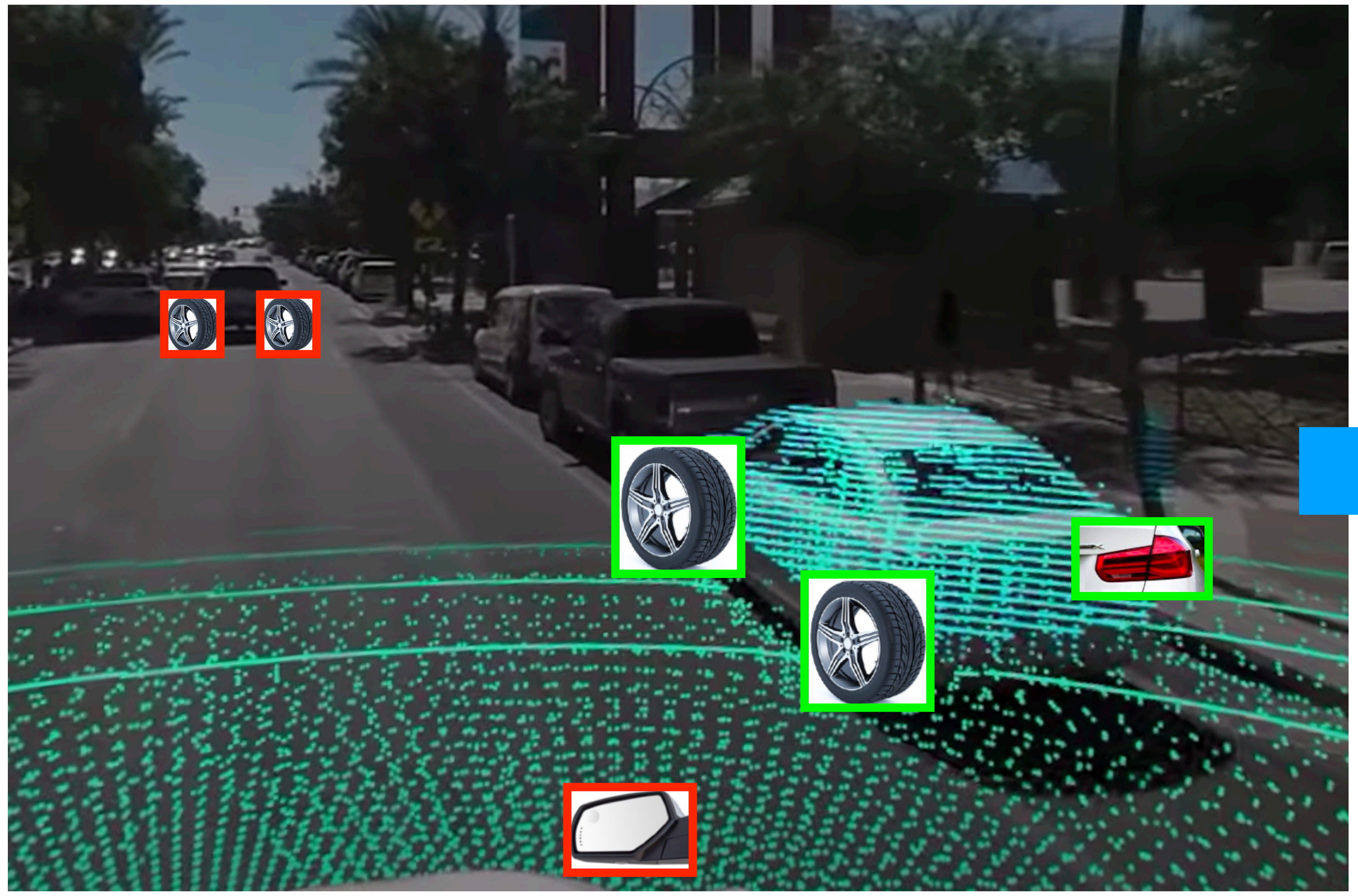
Fast multi-threaded open-source code:
https://github.com/MIT-SPARK/TEASER-plusplus

[Yang and Carlone. A Polynomial-time Solution for Robust Registration with Extreme Outlier Rates. RSS 2019]
[Yang, Shi, Carlone. TEASER: Fast and Certifiable Point Cloud Registration. TRO 2020]

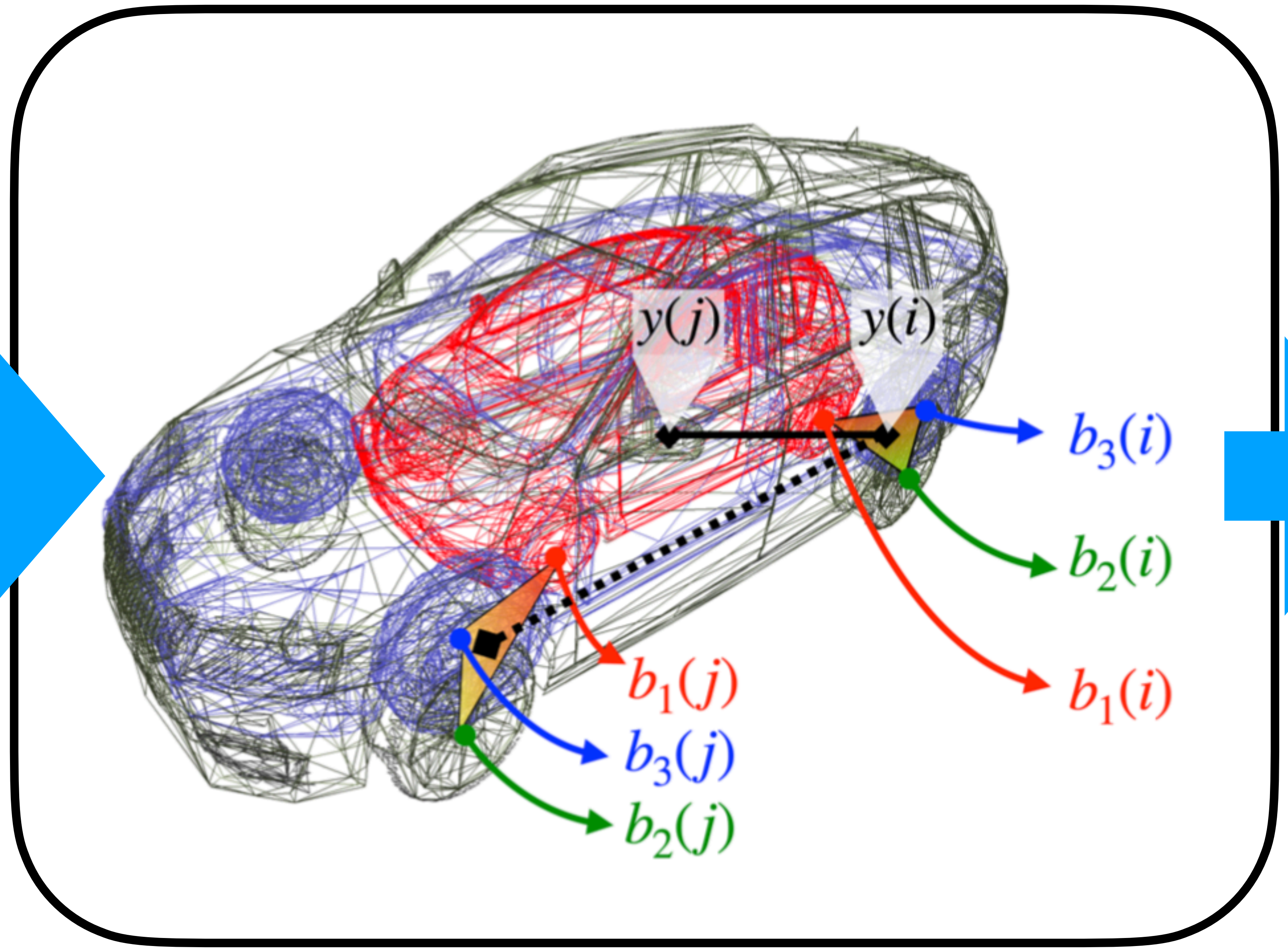
PACE#: category-level object localization in RGB-D images

Object To detect

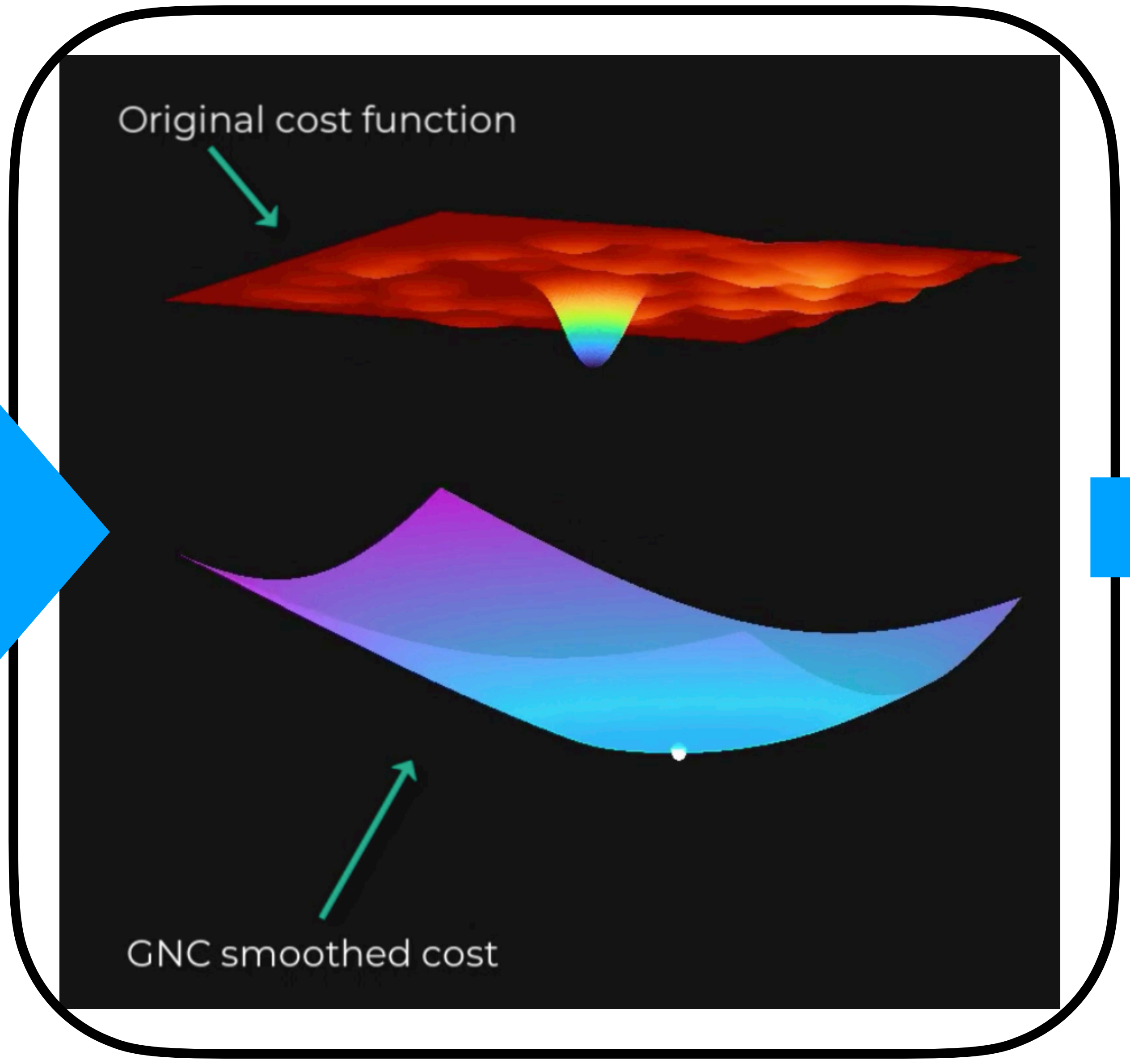
Sensor data



ROBIN



GNC

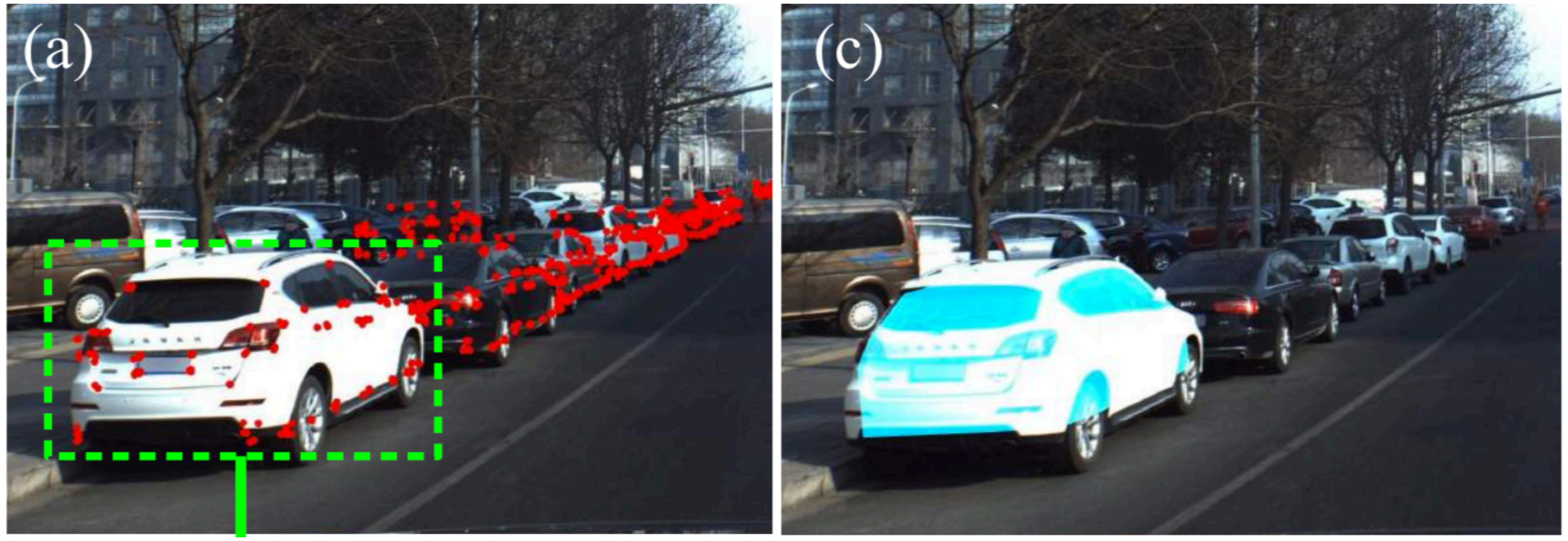


Pose And Shape

[Shi, Yang, Carlone. Optimal Pose and Shape Estimation for Category-level 3D Object Perception. RSS 2021]

PACE#: category-level object localization in RGB-D images

ApolloScape self-driving dataset: 200 images (validation split), 79 car models, 66 keypoints per car.

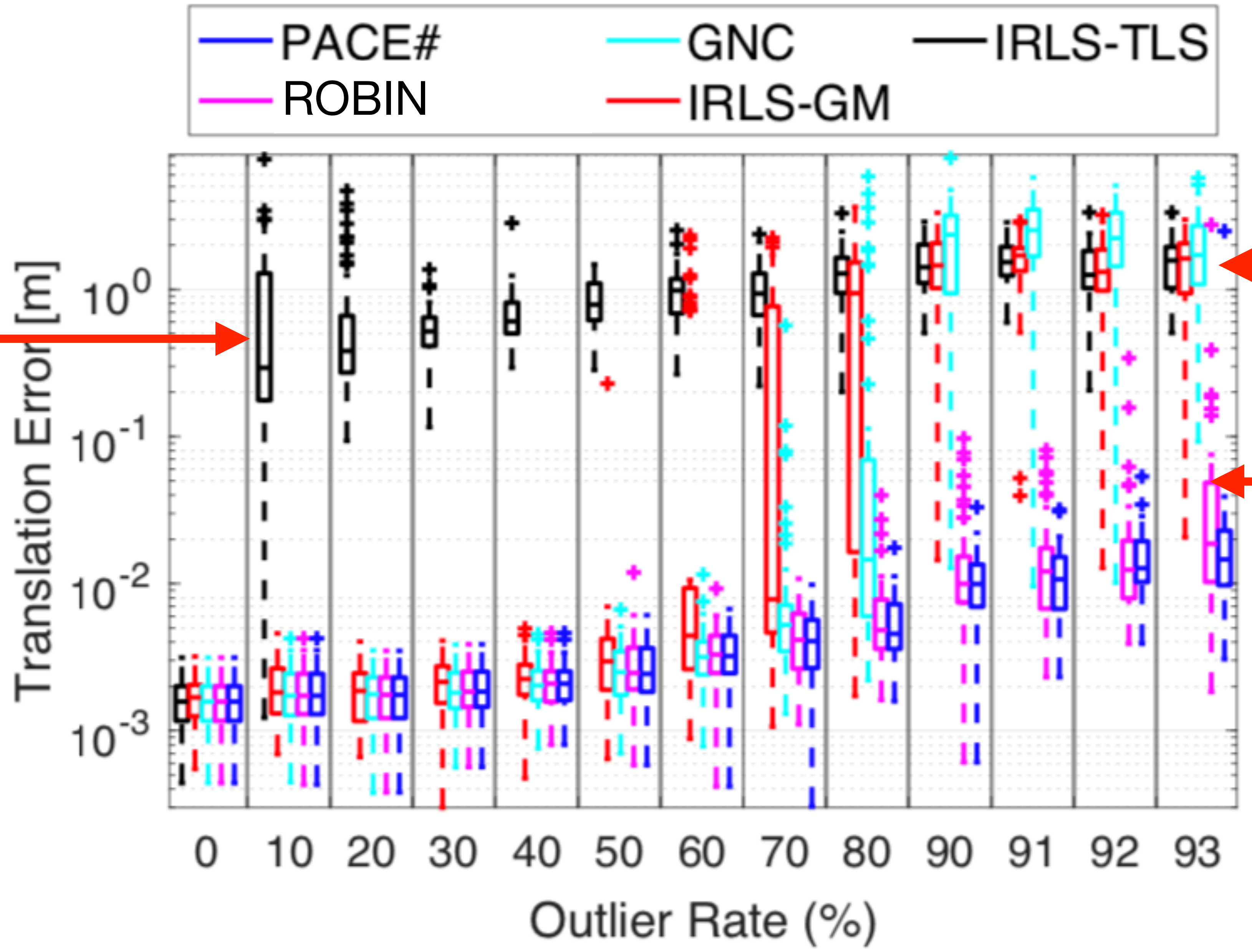


Deep learning baselines (2017-2020)

Strict criterion: % of estimates with <1.4m translation error and <15° for rotation error

	A3DP-Rel ↑			A3DP-Abs ↑		
	mean	c-l	c-s	mean	c-l	c-s
DeepMANTA [10]	16.0	23.8	19.8	20.1	30.7	23.8
3D-RCNN [34]	10.8	17.8	11.9	16.4	29.7	19.8
GSNet [30]	20.2	**40.5**	19.9	18.9	**37.4**	18.4
PACE#-ApolloDepths	**25.9**	35.7	**33.7**	**22.4**	34.7	**31.6**
PACE#-GTDepths	36.0	45.4	43.6	35.3	44.2	43.2
PACE#-GTKeypoints	64.5	88.1	86.0	64.3	88.1	86.1

Common robust estimators



GNC without ROBIN

ROBIN without GNC

PACE# (proposed)

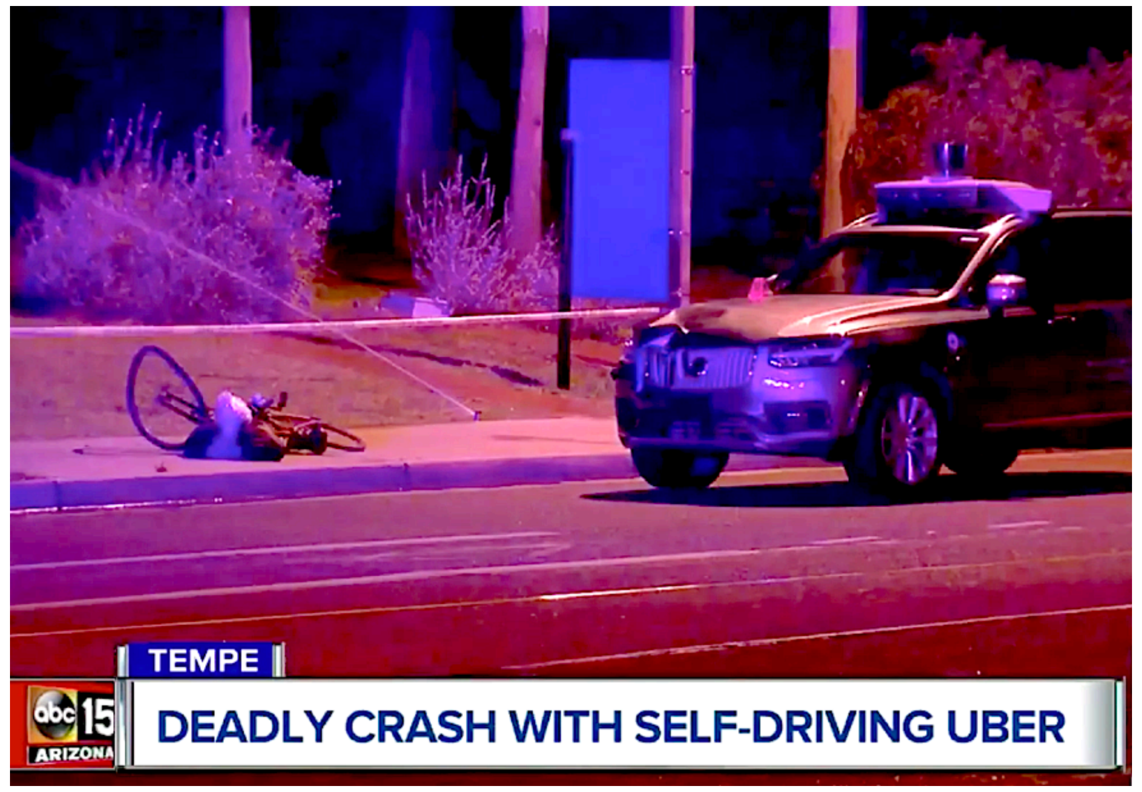
GSNet keypoint detection	PACE# Max-clique	PACE# GNC
0.45 s	2ms	0.45 s

TABLE II: Timing breakdown for PACE#.

Non-optimized Python implementation

[Shi, Yang, Carlone. Optimal Pose and Shape Estimation for Category-level 3D Object Perception. arxiv 2021]

Outline



What is a "certifiable algorithm" for geometric perception?

- Graduated Non-Convexity
  [ICRA'20, CVPR'20]
- Graph-theoretic outlier pruning
  [TRO'20, ICRA'21, arxiv'21]
- Optimality certification
  [TRO'20, NeurIPS'20]

Our goal: Certifiable Algorithms

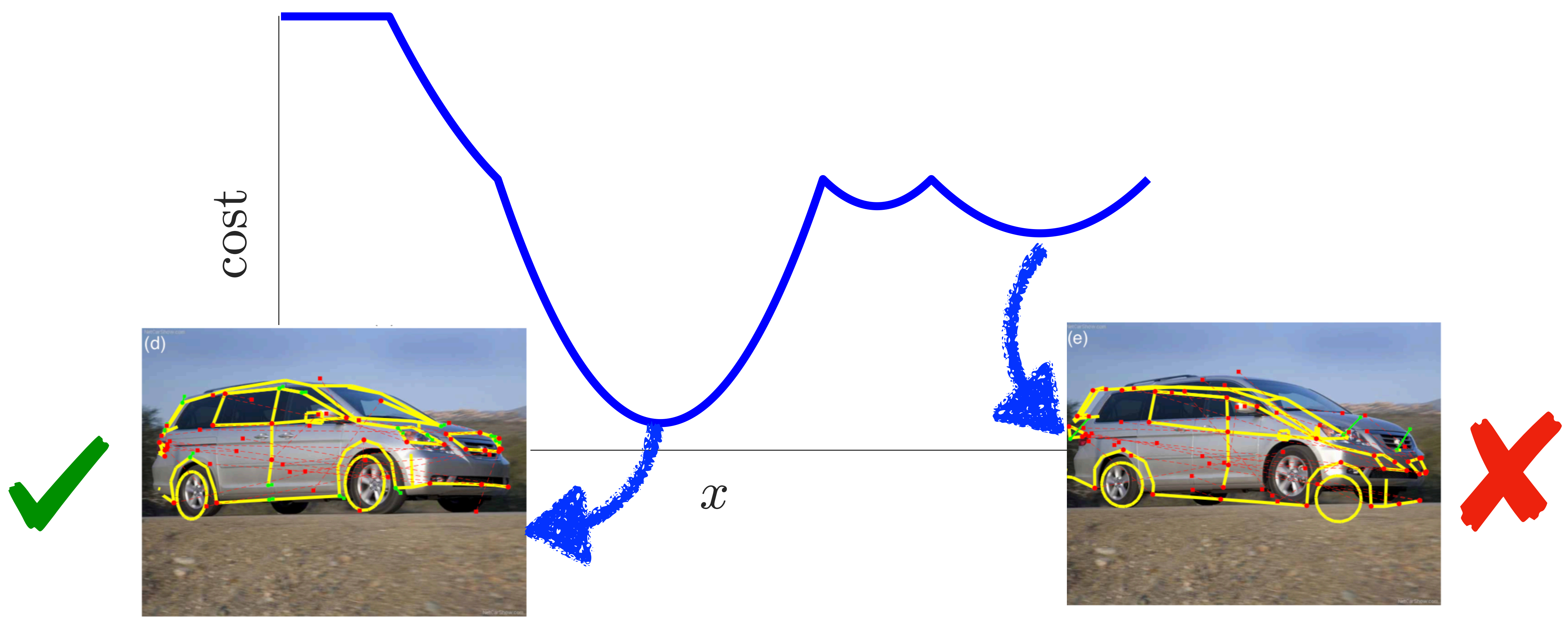
Input
(measurements y_i)

Algorithm

$$\min_{\boldsymbol{x} \in \mathbb{X}} \sum_{i \in \mathcal{M}} \rho(r(\boldsymbol{x}, \boldsymbol{y}_i))$$

Output
(estimate)

Certifiable algorithms: fast algorithms that (w/o requiring an initial guess) solve robust estimation to optimality in virtually all problem instances or detect failure when unable to compute an optimal solution



How to certify optimality of a given estimate?

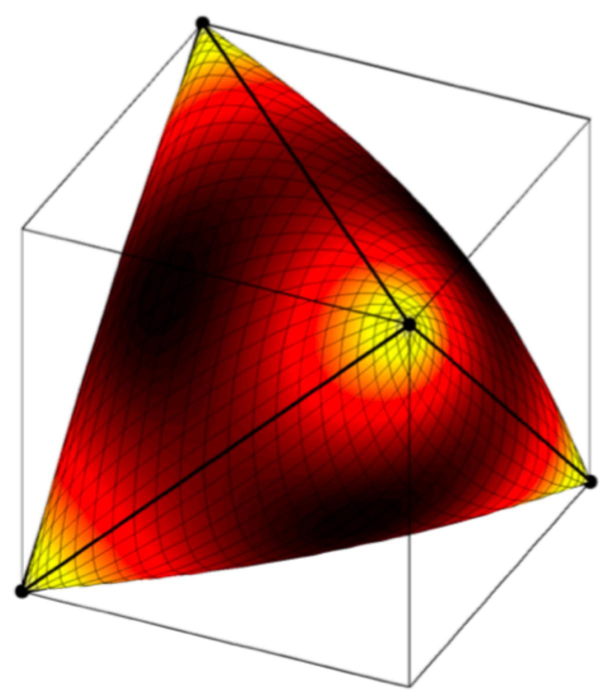
Robust Model Fitting
(non-convex, combinatorial)

$$\arg\min_{\substack{x\in\mathbb{X},\\ \theta_i\in\{0,1\},\forall i}} \sum_{i\in\mathcal{M}} \theta_i \|r_i(x,y_i)\|^2 + (1-\theta_i)\bar{c}^2$$

Polynomial Optimization
(non-convex, hard)

$$\begin{aligned} \min_{\boldsymbol{x}\in\mathbb{R}^n} \quad & f(\boldsymbol{x}) \\ s.t. \quad & h_i(\boldsymbol{x}) = 0, i = 1,\ldots,m, \\ & g_k(\boldsymbol{x}) \geq 0, k = 1,\ldots,l, \end{aligned}$$

Lasserre Hierarchy
(semidefinite programming, convex)



Optimality certificate

Theorem (informal): If a given estimate x is optimal for the convex relaxation, then it is also optimal for the original non-convex problem.

- Optimality of the convex relaxation can be checked in polynomial time
- Systematic way to derive a convex relaxation using Lasserre's moment relaxation
- Tight relaxation: x optimal for relaxation **if and only if** it's optimal for original problem

[Yang and Carlone, One Ring to Rule Them All: Certifiably Robust Geometric Perception with Outliers, NeurIPS'20.]

TEASER++: Estimation contract

- We can provide the first performance guarantees for geometric perception problems:

Theorem (Exact Recovery): if the measurements (i) contain at least 3 noiseless and non-collinear inliers, (ii) the outliers are not adversarial, and (iii) the certificate of optimality holds, TEASER++ recovers the true object pose.

Theorem (Exact Recovery in **Adversarial** Setting): if the number of noiseless and non-collinear inliers is larger than the number of outliers ($N_{in}>N_{out}$ + 3) and the certificate of optimality holds, TEASER++ recovers the true object pose.

- In practice: we can discard suboptimal estimates as incorrect:

Scenes	Kitchen (%)	Home 1 (%)	Home 2 (%)	Hotel 1 (%)	Hotel 2 (%)	Hotel 3 (%)	Study Room (%)	MIT Lab (%)
RANSAC-1K	90.9	91.0	73.1	88.1	80.8	87.0	79.1	81.8
RANSAC-10K	96.4	92.3	73.1	92.0	84.6	90.7	82.2	81.8
TEASER++	97.7	92.3	82.7	96.9	88.5	94.4	88.7	84.4
TEASER++ (CERT)	99.2	97.5	90.0	98.8	94.9	97.7	94.8	93.9

[Yang, Shi, Carlone. TEASER: Fast and Certifiable Point Cloud Registration. TRO 2020]

Summary

Measurements

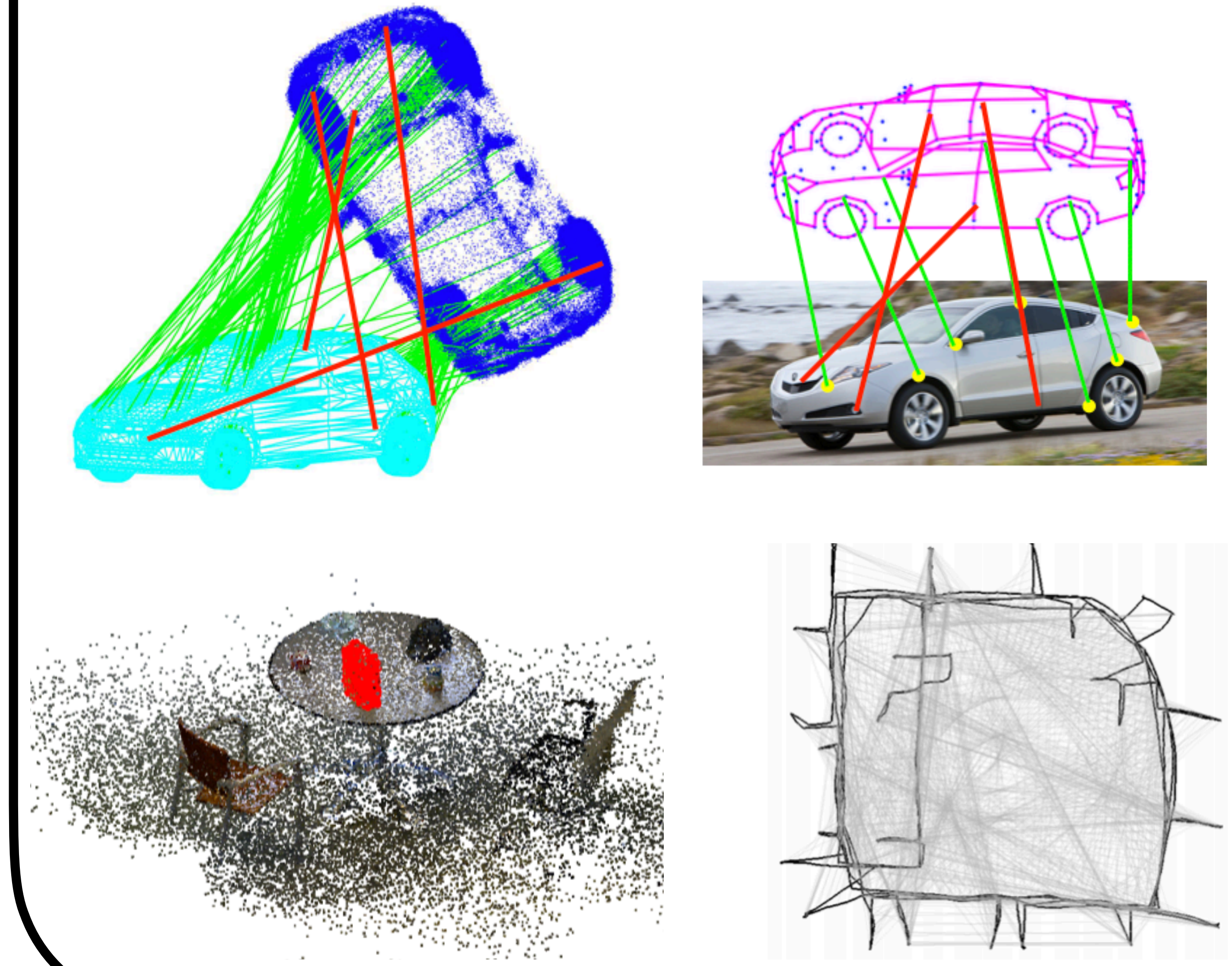
Subset of measurements

Robust Estimate

Optimality certificate

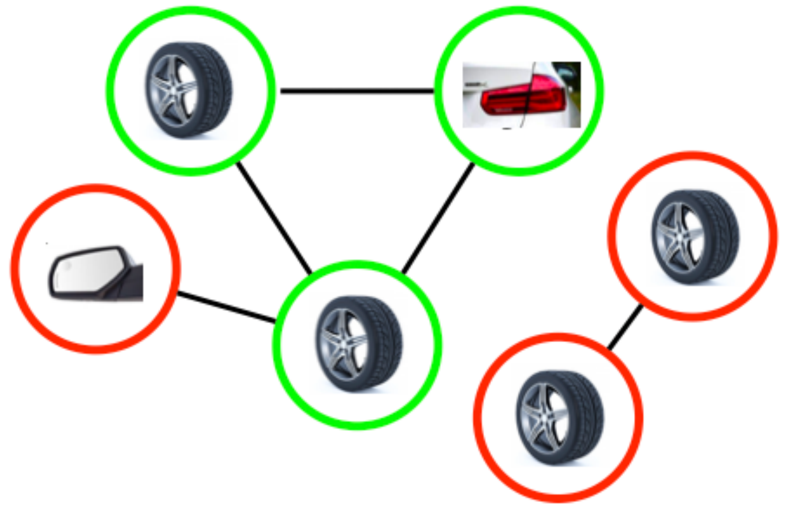
Truncated Least Squares (TLS) Robust Estimation

Combinatorial, Non-Convex, Hard



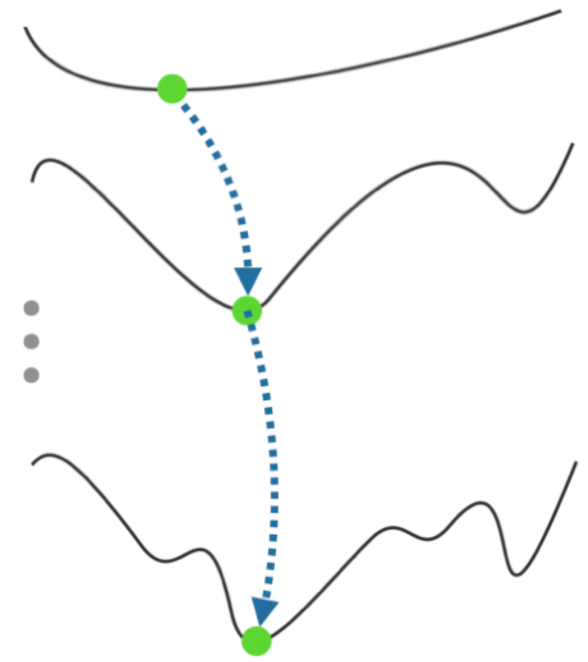
$$\underset{\substack{x\in\mathbb{X},\\ \theta_i\in\{0,1\},\forall i}}{\arg\min} \sum_{i\in\mathcal{M}} \theta_i \|r_i(x,y_i)\|^2 + (1-\theta_i)\bar{c}^2$$

ROBIN



Graph-theoretic outlier pruning

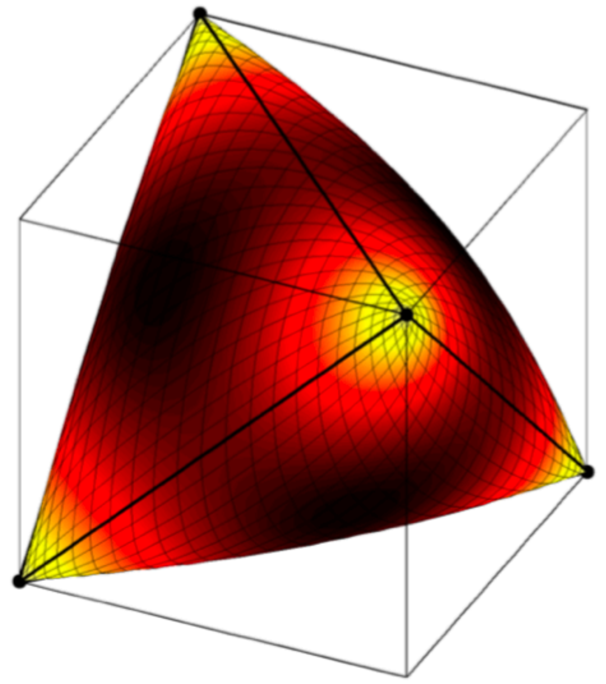
GNC



Graduated Non-Convexity

Certification

Convex relaxation



Minimum-order Moment/SOS Relaxation (Semidefinite Program)

Robust Estimate with certificate of optimality

[Yang and Carlone, One Ring to Rule Them All: Certifiably Robust Geometric Perception with Outliers, NeurIPS'20.]

NEW! Self-supervised Geometric Perception

Self-supervised!

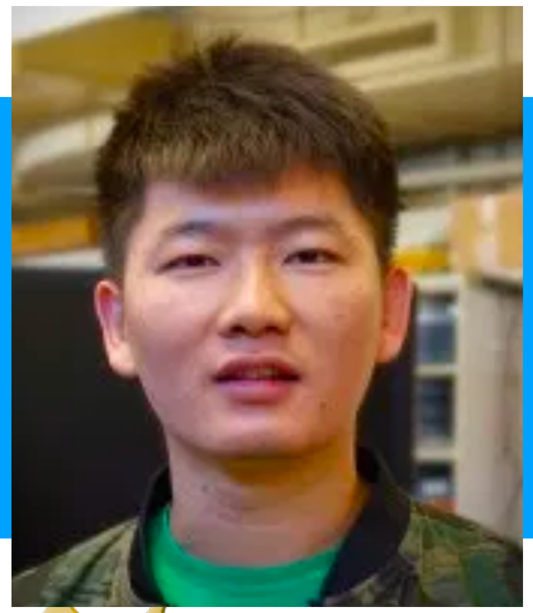
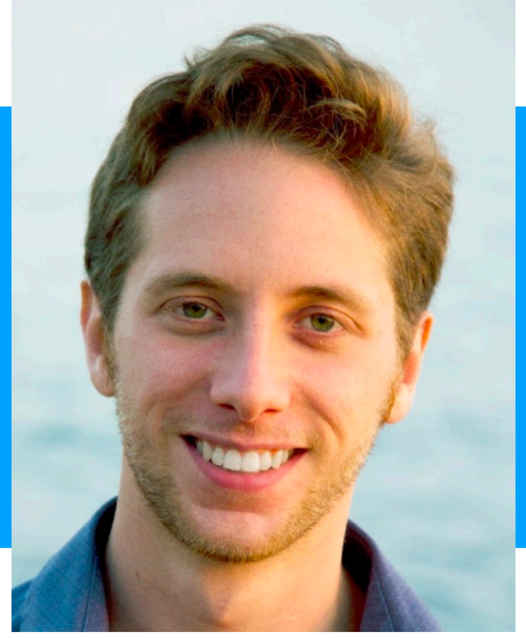
~~**Based on supervised learning:**~~
~~human annotates features and correspondences~~

Idea: why not using inlier/outlier decisions from robust estimation as a supervisory signal to deep feature learning?

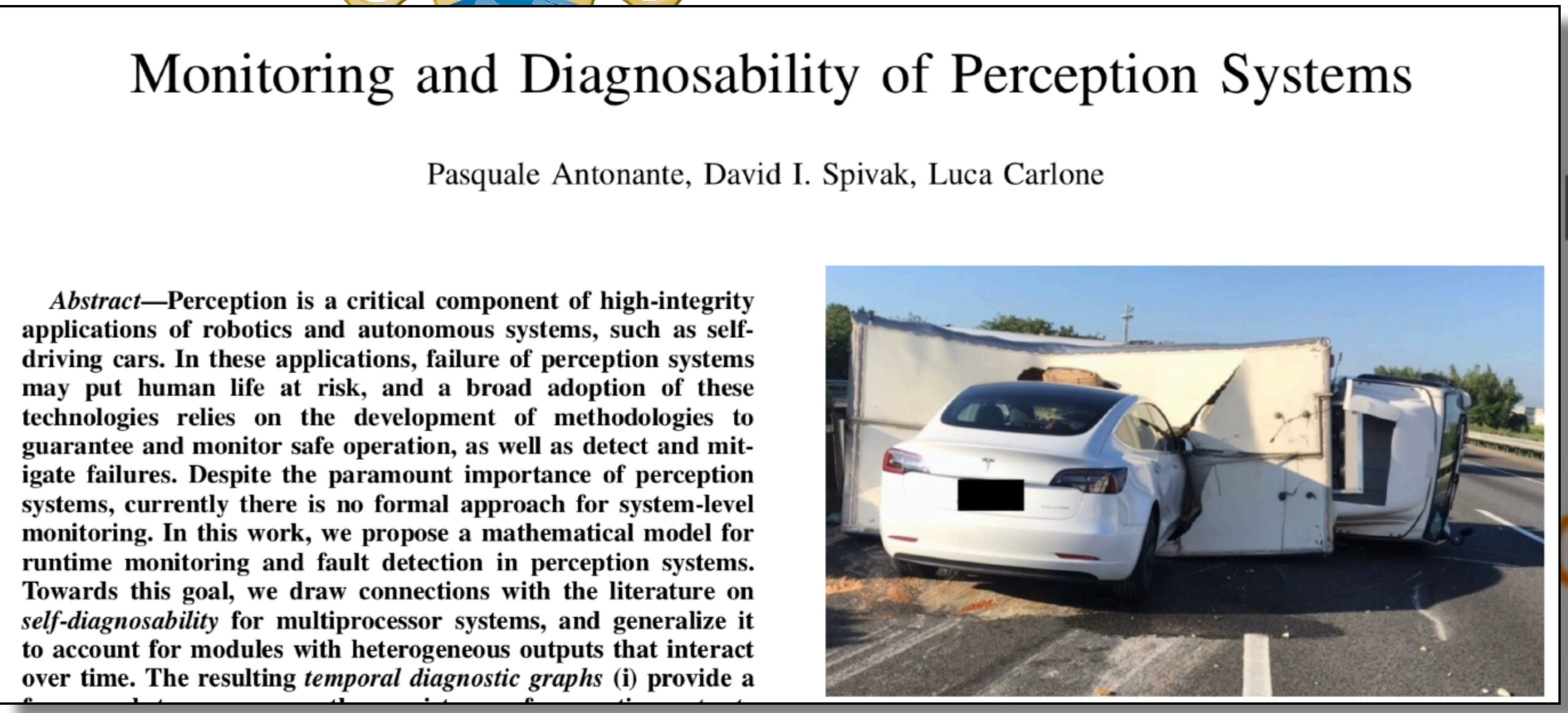
[Yang, Dong, Carlone, Koltun, Self-supervised Geometric Perception. CVPR'21]

Conclusion

- New opportunities to improve robust estimation techniques for perception:
  - ➡ **Certifiable algorithms:** solve optimization globally without initial guess and certify optimality (or detect failures)
  - ➡ Graduated non-convexity, graph-theoretic outlier pruning, certification
- Many other questions: how to certify the front-end? system-level guarantees?

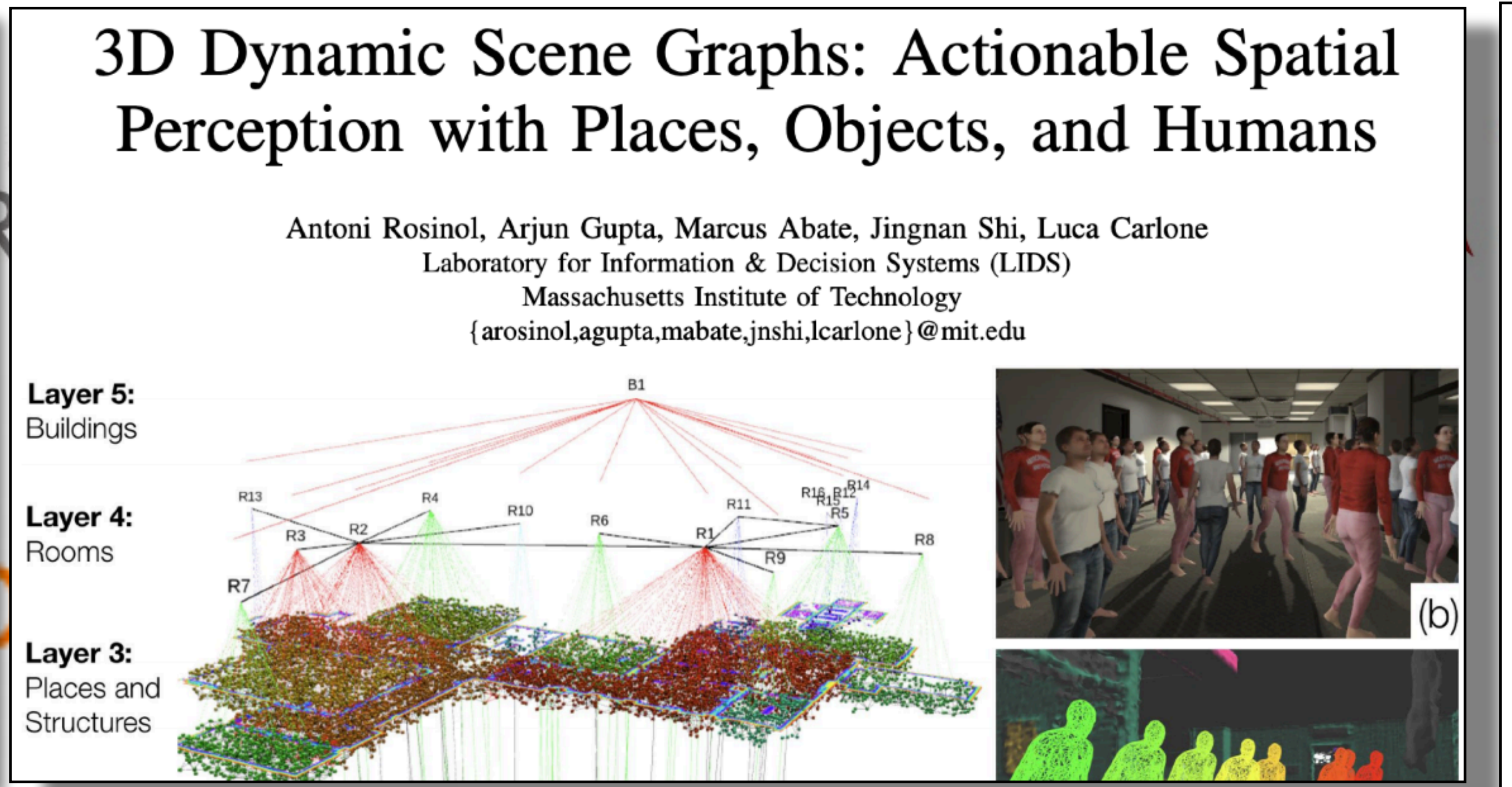
Thank you!

Monitoring and Diagnosability of Perception Systems

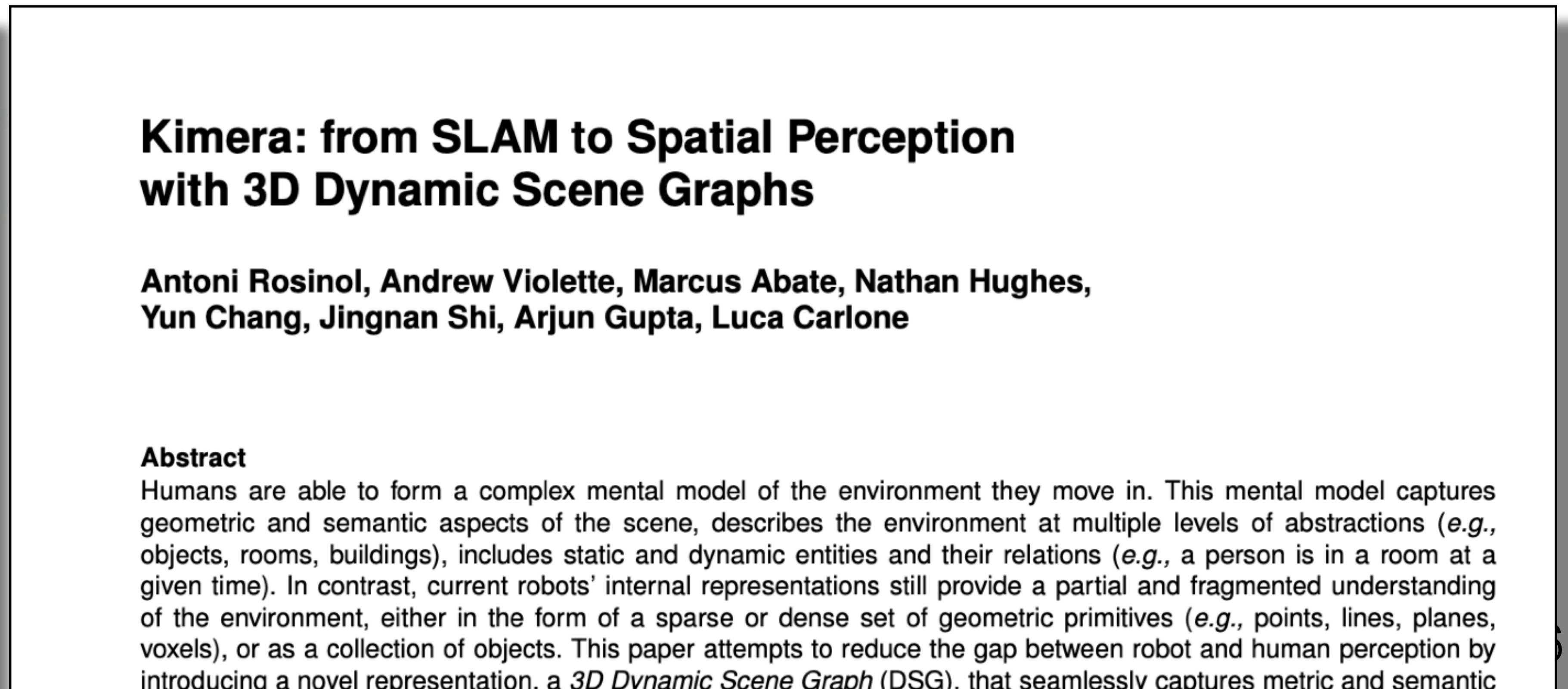
Pasquale Antonante, David I. Spivak, Luca Carlone

Abstract—Perception is a critical component of high-integrity applications of robotics and autonomous systems, such as self-driving cars. In these applications, failure of perception systems may put human life at risk, and a broad adoption of these technologies relies on the development of methodologies to guarantee and monitor safe operation, as well as detect and mitigate failures. Despite the paramount importance of perception systems, currently there is no formal approach for system-level monitoring. In this work, we propose a mathematical model for runtime monitoring and fault detection in perception systems. Towards this goal, we draw connections with the literature on *self-diagnosability* for multiprocessor systems, and generalize it to account for modules with heterogeneous outputs that interact over time. The resulting *temporal diagnostic graphs* (i) provide a



3D Dynamic Scene Graphs: Actionable Spatial Perception with Places, Objects, and Humans

Antoni Rosinol, Arjun Gupta, Marcus Abate, Jingnan Shi, Luca Carlone
Laboratory for Information & Decision Systems (LIDS)
Massachusetts Institute of Technology
{arosinol,agupta,mabate,jnshi,lcarlone}@mit.edu



Kimera: from SLAM to Spatial Perception with 3D Dynamic Scene Graphs

Antoni Rosinol, Andrew Violette, Marcus Abate, Nathan Hughes, Yun Chang, Jingnan Shi, Arjun Gupta, Luca Carlone

Abstract
Humans are able to form a complex mental model of the environment they move in. This mental model captures geometric and semantic aspects of the scene, describes the environment at multiple levels of abstractions (*e.g.,* objects, rooms, buildings), includes static and dynamic entities and their relations (*e.g.,* a person is in a room at a given time). In contrast, current robots' internal representations still provide a partial and fragmented understanding of the environment, either in the form of a sparse or dense set of geometric primitives (*e.g.,* points, lines, planes, voxels), or as a collection of objects. This paper attempts to reduce the gap between robot and human perception by introducing a novel representation, a *3D Dynamic Scene Graph* (DSG), that seamlessly captures metric and semantic