IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2024.0429000

Applications of Robust Statistics in Autonomous Driving

TINO WERNER[1]
[1]German Aerospace Center (DLR), Institute of Systems Engineering for Future Mobility, Oldenburg, Germany (e-mail: tino.werner@dlr.de)
Corresponding author: Tino Werner (e-mail: tino.werner@dlr.de).

The research leading to these results is funded by the German Federal Ministry for Economic Affairs and Climate Action within the project "KI Wissen – Entwicklung von Methoden für die Einbindung von Wissen in maschinelles Lernen". The author would like to thank the consortium for the successful cooperation. This work also received funding by the V&V4NGC project of the German Aerospace Center.

ABSTRACT Autonomous vehicles have to interact with their environment with the goal to fulfill their tasks while respecting all desired constraints such as not causing dangerous situations, driving comfortable maneuvers, enabling a smooth traffic flow, or avoiding overly polluting driving behavior. All steps require a suitable perception of the environment conditions, such as the estimation of the own position, a prediction of the trajectories of other traffic participants, or the assessment of parameters corresponding to vehicle dynamics. However, classical estimation algorithms are known to be easily distorted by outliers in the data. In addition, apart from rule-based systems, it becomes more convenient to train autonomous agents by machine learning algorithms. Again, such algorithms need to be robust in order to cope with model misspecification or outliers in the data. Robust Statistics is a discipline of statistics which exactly addresses these challenges. This paper provides an extensive and systematic overview of current applications of Robust Statistics in autonomous driving in a unified notation, discusses different notions of the term "robustness" and identifies directions for future work.

INDEX TERMS Autonomous Driving, Contaminated Data, Outliers, Robust Statistics

I. INTRODUCTION

Autonomous vehicles in operation have to interact with their environment by repeatedly successfully performing three tasks: Perceiving the current environment state, predicting the future states of all relevant traffic participants up to some prediction horizon, and planning their own maneuvers and therefore necessary control actions. Apart from rule-based systems, which operate according to a deterministic plan, works such as [19] have demonstrated that an autonomous agent can also be trained via machine learning (ML), here Imitation Learning (IL).

A major drawback of IL is the necessity to provide expert trajectories according to which the agent is trained. An alternative class of algorithms is given by Reinforcement Learning (RL), where no training data are required but where the agent learns by trial-and-error. However, while knowledge about correct maneuvers is implicitly encoded in the expert trajectories in IL training, and while traffic rules can be implemented in rule-based systems, RL agents learn according to a reward function which assigns a real value to a state-action pair, so that the agent learns by experience which actions were useful (i.e., resulted in higher rewards) for which states. Projects such as KI Wissen[1] consider the formalization of prior knowledge and their integration into Artificial Intelligence (AI) training for autonomous driving, see [331] for an extensive overview of knowledge integration into AI.

The perception, the prediction and, for an agent trained by machine learning, even the training is based on statistical estimation, which is well-known to be vulnerable to contamination of the data in the sense that the true model differs from the assumed model, the "ideal model", so that observations from the real distribution may appear as outliers (w.r.t. the "ideal model"), with the potential to severely distort a statistical estimator (see, e.g., [153], [121], [213]). Robust Statistics has provided numerous techniques in order to safeguard against such perturbations in the sense that the estimator still works reasonably well, even in the presence of a certain fraction of contaminated data.

Due to the rising interest in autonomous systems and the constant progress made in robustifying estimation and machine learning algorithms, this paper aims at systematically collecting concrete applications of Robust Statistics in autonomous driving, to formalize these approaches in a

[1]https://www.kiwissen.de/

consistent mathematical notation, and to identify possible extensions and directions for further research.

This paper is organized as follows. Sec. II provides a description of each tasks considered in this paper, a placement into the "sense, plan, act" workflow, and a roadmap across the different application areas considered in this paper. Moreover, potential sources of contamination are identified and how such contamination appears in the data. Sec. III collects the necessary concepts from Robust Statistics and relates them to other notions of robustness that one encounters in the autonomous driving literature. Sec. IV is devoted to approaches from Robust Statistics in perception tasks such as tracking, point cloud detection, or state estimation. Sec. V reviews robustifications for planning/prediction tasks, in particular for RL, IL, and model-predictive control (MPC) algorithms. In Sec. VI, potential topics for future work are discussed.

II. OVERVIEW

A. GENERAL TASKS

An autonomous vehicle has to continuously observe its surroundings (perception, "sense"). This is realized in practice by potentially multiple types of sensors such as cameras, LiDAR or radar sensors. The collected information is used in order to predict the movements of the surrounding traffic participants, such as other vehicles or pedestrians, which is necessary in order to plan its own maneuvers ("plan"/"think"). The planning outcomes are finally used in order to perform the correct actions so that the planned next state is reached ("act"). An graphical illustration, including selected tasks corresponding to each of these phases, is provided in Fig. 1.

In the following subsections, we briefly describe each task which has already been addressed by techniques from Robust Statistics that we review in greater detail later. They should provide a quick overview for the reader and already collect the main challenges corresponding to the respective task concerning sources of contamination and the impact of contamination on inference and optimization. Mathematical formulations of the respective optimization problems and the methodology are postponed to the main sections Sec. IV and Sec. V.

B. SIMULTANEOUS LOCATION AND MAPPING

Simultaneous location and mapping (SLAM) consists of two main tasks: Tracking the position of the robot (strictly speaking, the sensor) and estimating its ego-motion; and computing the map of the unknown surrounding environment (e.g., [168]). The robot may use different types of sensors such as camera, LiDAR, sonar or infrared. Camera-based SLAM is referred to as visual SLAM (e.g., [353]).

In odometry, the goal is to estimate the ego-motion of the robot. In contrast to SLAM, which requires global consistency of the estimated trajectory in regard of a localization of the agent within its environment, odometry considers local consistency and incrementally estimates the robot's trajectory. Odometry can be considered as part of SLAM, e.g., [128].

In particular, one has to distinguish between different types of sensors that are used for odometry, e.g., wheel odometry, GNSS/INS, GPS, sonar, LiDAR or camera (e.g., [13]). Using camera data corresponds to visual odometry (VO), using LiDAR data to LiDAR odometry (e.g., [177]).

Depending on the used sensor type and the actual task, one may find different types of contamination in the collected data, which we specify in the following subsections.

1) Visual odometry

As for camera models in VO, the most common is the perspective camera model (e.g., [353]). All camera models map the 3D world into an 2D image plane. In the perspective model, more distant objects appear smaller. For 2D image coordinates (u, v) and 3D coordinates (x, y, z), the perspective model is given by

$$\lambda \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix},$$

where λ is a depth factor and where the matrix is referred to as the intrinsic calibration matrix, with the focal lengths f_x and f_y and the projection center (c_x, c_y).

In general, cameras are vulnerable to illumination changes (e.g., [63]). Other sources of contamination can be self-shadowing, camera saturation, camera shaking or rotation, motion blur or defocus ([219]).

In direct odometry, the image data are used as they are and a projection of the images on reference images is computed. The quality is then quantified via the photometric errors [125]. Alternatively, [62] use brightness intensities at the positions instead of the position coordinates themselves. In the sample consisting of the collected 2D points, contamination manifests itself in points that are not in accordance with the rest of the sample, resulting in high photometric and/or geometric errors when comparing the source and the reference image. A robust approach allows to cover situations (regardless whether photometric or geometric errors are quantified) where the image taken by the camera is contaminated, but in principle, it would also allow for the usage of contaminated reference images or reference intensities.

[219] mention that a stereo camera pair increases the robustness for the cost of slightly increases computational complexity.

Optical flow estimation slightly differs from VO since the only goal is to estimate the optical flow, i.e., the velocity between subsequent images, but not necessarily the camera position itself. According to [155], the optical flow can be related to the position, translational and rotational velocity of the camera. Therefore, one can extract the positions and velocities by regression from the sample. Contamination in the sample is induced by measurement errors of the optical flow, maybe due to the lack of a visible ground surface. [33], [34] elucidate that optical flow estimation is usually accompanied by assumptions such as the brightness constancy assumption which indicates that the brightness only

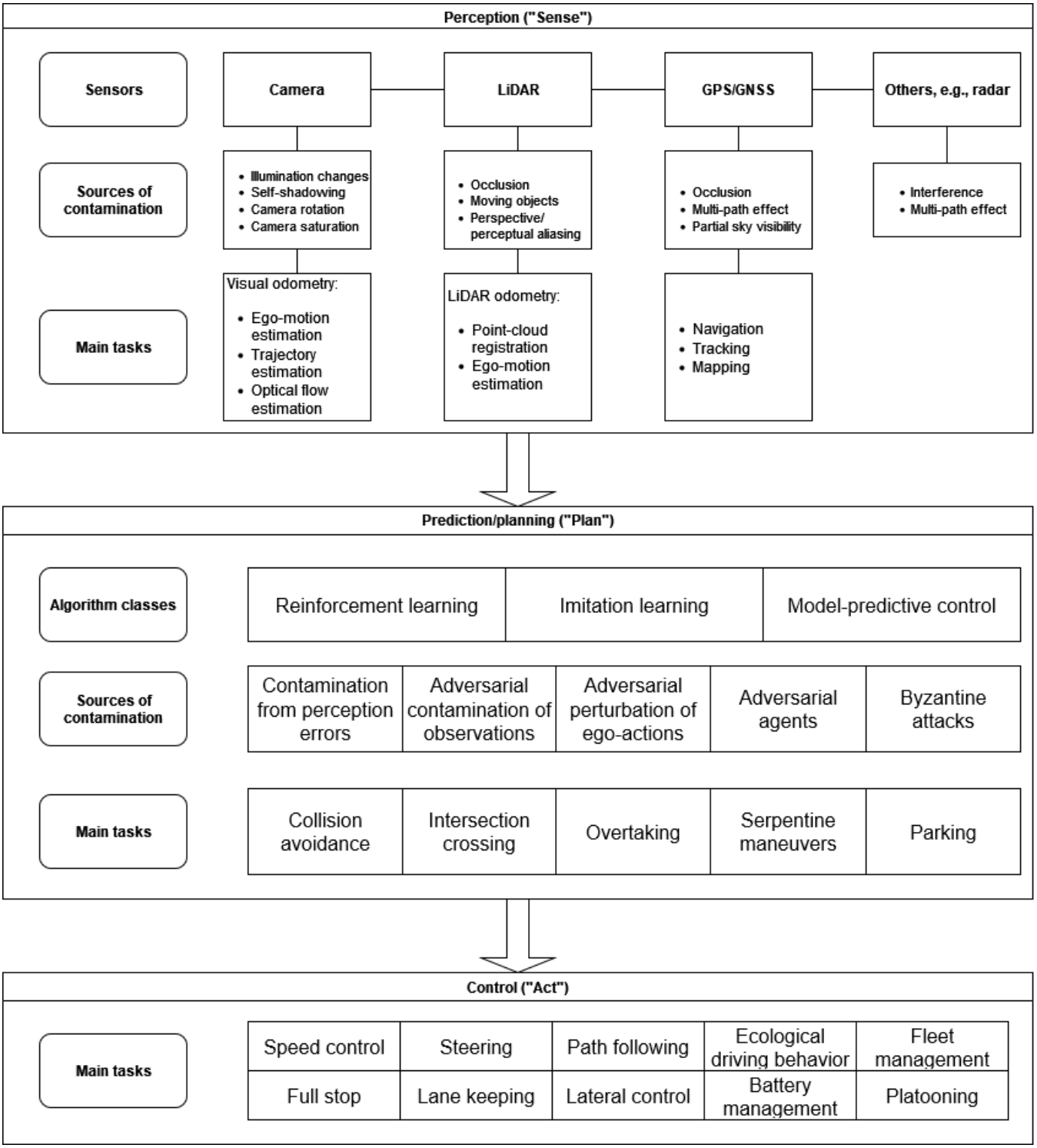

FIGURE 1. Overview of perception, planning, and control tasks as well as corresponding sources of contamination.

varies smoothly (w.r.t. to both position and time), or a spatial coherence assumption that indicates that neighboring pixels are likely to be part of the same object, and hence change similarly. Those assumptions however are violated in the presence of reflections, shadows, or motion boundaries, which imply outliers in the data in the sense that they violate these assumptions. In other words, outliers are likely to produce large loss values and therefore capable to let the estimation break down. Therefore, robust approaches are required.

2) LiDAR odometry / point cloud registration

LiDAR data are usually point clouds. In contrast to cameras, LiDAR has the advantage to be immune to light variations [177], but the analysis of 3D data may cause high computational costs [13]. When using LiDAR data, contamination may result from occlusions [381], moving objects [62], [63], perspective and perceptual aliasing, i.e., different but similar places cannot be distinguished [210], or from environments with self-similar structures where false associations are generated that appear as outliers [4]. [245] argue that outliers not only occur due to measurement errors but also due to changes in the object itself or overlapping point clouds in the sense that they represent overlapping but not the same details of an object. [330] point out that moving objects appear as dynamic obstacles and hence occlude the static environment from the sensors of an autonomous vehicle, inducing outliers. See [187] for an overview of potential sources of contamination in point cloud data.

Point cloud registration addresses the registration of sets of 2D or 3D points in computer vision. One of the best known algorithms for this problem is the iterative closest point algorithm (ICP). Here, one has some reference surface (a "model" point cloud), with which the observed point cloud, the "data" point cloud, has to be aligned. To this end, the goal is to find a parametric transformation between the points of both point clouds. In the simplest form, this is done iteratively by finding the closest model point for each data point, respectively, for the current parameter, and to update the transformation parameter by minimizing the sum of squared distances over all data points (e.g., [93]). Note the similarity of this approach to transform data points in order to match model points and the transformation of 2D positions or brightness values in order to match their reference counterparts in VO.

Such outliers potentially induce large values of the loss function that is to be minimized, in particular, when using the squared loss as in the standard ICP algorithm. A typical robustification is to use a robust loss function that allows for a certain fraction of such erroneous points without significant distortion of the matching procedure.

3) Navigation

For satellite data, contamination may occur due to occlusion, i.e., where the line-of-sight between emitter and receiver is blocked [237], [231], [269]. [9] argue that a common source of contamination in GNSS data is the multi-path effect, i.e., the signals from satellites in a low orbit can reach the receiving antenna by multiple paths due to reflection on the ground or on surrounding objects. [65] additionally mention partial sky visibility and non-line-sight as sources of contamination. Another source of contamination can be electromagnetic propagation in the sensors [334]. Radar data may be contaminated due to the multi-path effect and interference ([192]).

In odometry, although the models may receive sequential data, they usually do not include a time component in the form of a time series or state space model. All navigation or tracking approaches where time series data are used are discussed separately and are therefore contained in the individual section Sec. IV-A3. The often used filtering approaches assume a state space model where the observations at time t depend on the current system state, and the current system state depends recursively on the system state at time $(t-1)$. In contrast to localization problems that are solved via regression or image translation where contamination appears statically in the responses, regressor matrices, or point clouds/images, contamination can propagate through the recursive state space model, and the error distribution may be non-stationary.

Usually, one assumes outliers that only affect the observations, indicating that the observations can be drawn from a different than the assumed ideal distribution with a certain probability (cf. [269]). However, contamination may also appear in the state equation, indicating that the state at time $(t+1)$ is drawn from a different than the assumed conditional distribution w.r.t. the current state at time t. This phenomenon is referred to as innovation outliers (cf. [269]), and results, due to the recursivity of the state equation, also in contamination in the subsequent states.

Filtering approaches mostly amount to the application of the standard or an extended Kalman filter (which covers non-linear state and observation equations), or by noise modeling, usually by variational inference. Inference in Kalman filtering is often done via least squares regression, opening the path for robust approaches by performing robust regression instead. In variational inference, assumptions on the measurement and process noise distributions are required. By using a heavy-tailed distribution for each type of noise, both additive and innovation outliers can be adequately captured when computing the posterior state distributions.

[343] point out that many approaches try to achieve robustness by focusing on high-level features such as lines and edges, however, the computational burden can hinder real-time performance. [211] point out that robust features (e.g., [203], [27]) that are constructed in order to be less invariant towards illumination changes are not suitable for realistic situations where the spectrum and the direction of the light can change.

In SLAM, the data may consist of relative information such as (pseudo)range measurements, i.e., distances between the receiver and the emitter, collected from GPS or radar. By a regression model, one can infer the vehicle (i.e., receiver) location from the range data. Since the measurement function that quantifies the relative information is usually non-linear, a good initial guess for the true positions is required in order to find the global optimum (cf. [141]). Contamination appears not only from measurement errors but is also implied by bad initial guesses, and their location may be random or grouped (e.g., [4]). Contamination in the sense of measurement errors of the ranges appear as outliers in the responses. A robust (regression) approach therefore not only safeguards against measurement errors but also against bad initializations (maybe resulting from measurement errors in the data) and allows for contamination in the regressor matrix, maybe due

to erroneous receiver coordinates or clock offsets.

C. BOUNDING BOX ESTIMATION

Bounding box estimation is crucial in object detection and tracking. The goal is to find a box which completely surrounds the desired object and which is as tightly as possible. Typical approaches for bounding box detection invoke both a classification loss, because the object corresponding to the box needs to be identified, as well as a regression loss that quantifies the quality of the bounding box coordinates (e.g., [108]).

A source of contamination could be errors made by the annotators, which is a known problem and has been studied for example in [306] and [224]. Suppose that an annotator provided a bounding box that is much wider or narrower than it should be. This could induce wrong patterns during training, maybe when there are many similar objects in the data, so that the NN learns to fit reliable bounding boxes for the similar objects and, therefore, also for the object with the wrong bounding box, which implies a large regression loss for this particular bounding box. However, if for some reason a reference bounding box immensely differs from the optimal one, it could have a leverage effect, similarly as one large outlier in least-squares regression, and distort the whole model. Having already annotated data, is seems implausible that they would be checked again by the data scientist, so in this case, one could find contamination by inspecting the losses during NN training or during testing, and may identify such "outliers". By a robust approach, even large outliers may not result in a distorted model. [154] provided another argumentation why contamination may appear, namely that object detection is accompanied with an uncertainty that grows as a function of time, i.e., when performing tracking and iteratively predicting a bounding box for a future time step. In this sense, the true bounding box coordinates could appear as outliers under the assumption of the current predictive model for the coordinates.

D. ESTIMATION OF VEHICLE PARAMETERS

Vehicle parameters need to be inferred in order to operate a vehicle safely. Such parameters can correspond to vehicle dynamics like longitudinal and lateral velocities, moments of inertia, or tractive forces, or be related to electric vehicles only, which are voltages or the state of charge of batteries.

Contamination appears once measurement errors occur, either in the training data for the parameters of interest, resulting in errors in the response variables, or for the regressor variables, which leads to contamination in the features. Outliers in the responses or regressors can be identified individually by outlier detection procedures, applied onto the response column or the regressor matrix, however, in a regression setting, both procedures alone would not find outliers that are inconsistent with the regression model. In other words, if for example a response variable has been measured wrongly but still lies within the range of the majority of response values, it would not appear as outlier when considering the response column alone. However, if the contamination is inconsistent with an assumed underlying regression model, it is detected when computing the residuals. Robust procedures hence allow to deal with even large measurement errors, which may never be completely avoidable, and can provide models that only marginally suffer from contamination.

E. DETECTION OF ROAD FEATURES

Road feature detection includes to find the position of road features such as road markings or to even extract road surfaces from measured 3D point clouds. This is achieved by a regression approach.

A typical source of contamination are measurement errors and false positives of the tracker (e.g., [302]). In road surface detection, contamination may arise from non-surface points in the point cloud [234], which would imply large residuals.

F. PREDICTION AND PLANNING

Apart from the applications of Robust Statistics in perception tasks as illustrated in the previous subsections, robust procedures also entered approaches for prediction and planning, i.e., RL, IL, and MPC.

In most of these approaches, one either considers adversarial robustness or robustness against noise induced by measurement errors, where either the observed state is perturbed (e.g., [282]) or even the true state [316]. Robust approaches usually amount to a minimax game, where one does not optimize the expected future reward, but a worst-case future reward under unfavorable transitions [131], maybe induced by adversarial agents that are trained in order to hinder the ego-agent to reach its goal (e.g., [256]). Even perturbations of the ego-actions have been considered [298].

However, as RL does not need data but uses the data generated during training, finding a reasonable amount of contamination is more difficult than in settings where one can just use contaminated real-world data. One challenge, when training adversarial agents, is to ensure that these agents at least behave plausibly, i.e., that the ego-agent is not trained solely on extreme edge cases that are very unlikely to be encountered in the real world.

In IL, reference trajectories are given, where contamination can appear by random perturbations [19]. With annotation errors from bounding box estimation in mind, one can interpret such perturbations as a manifestation of a non-perfect human driver, similarly to annotation errors due to non-perfect human annotators. Alternatively, single state-action pairs may be contaminated [198], a fraction of transitions [366] or even a fraction of transitions and rewards [365].

III. KEY CONCEPTS OF ROBUST STATISTICS

This section provides the necessary notions and concepts of Robust Statistics.

A. CONTAMINATION MODELS

Contamination models formalize mismatches of the assumed ("ideal") and the real distribution. They are given by sets of

probability distributions whose distance, quantified in some metric, to the ideal distribution is bounded by some constant.

Before we start with formal definitions, we provide small examples in order to illustrate how the contamination models have to be understood and how they deviate from adversarial attacks.

Example 3.1: Consider the regressor matrix $X = \begin{pmatrix} 2 & 0 & 3 \\ -1 & 1 & 2 \\ -2 & 1 & 0 \\ 4 & -1 & 3 \end{pmatrix}$ and the following perturbed versions:

$$X^{(1)} = \begin{pmatrix} 1 & -1 & 2 \\ -2 & 0 & 1 \\ -3 & 0 & -1 \\ 3 & -2 & 2 \end{pmatrix}, \quad X^{(2)} = \begin{pmatrix} 8 & -4 & 6 \\ -1 & 1 & 2 \\ -2 & 1 & 0 \\ 4 & -1 & 3 \end{pmatrix}, \quad X^{(3)} = \begin{pmatrix} 2 & 0 & 3 \\ 8 & 9 & 8 \\ -2 & 1 & 0 \\ 9 & 8 & 11 \end{pmatrix}.$$

Let us start with adversarial attacks. Here, we interpret the perturbed matrices as sums of the form $X^{(k)} = X + V^{(k)}$, $k = 1, 2, 3$, for perturbation matrices $V^{(k)}$. These matrices are therefore given by

$$V^{(1)} = \begin{pmatrix} -1 & -1 & -1 \\ -1 & -1 & -1 \\ -1 & -1 & -1 \\ -1 & -1 & -1 \end{pmatrix}, \quad V^{(2)} = \begin{pmatrix} 6 & -4 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad V^{(3)} = \begin{pmatrix} 0 & 0 & 0 \\ 9 & 8 & 6 \\ 0 & 0 & 0 \\ 5 & 9 & 8 \end{pmatrix}.$$

Can these perturbation matrices stem from an adversarial attack scheme? It depends on the budget, usually quantified in the Frobenius norm. The Frobenius norm of a matrix $M \in \mathbb{R}^{m \times n}$ is defined as $||M||_F = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |m_{ij}|^2\right)^{1/2}$. Therefore, we have $||V^{(1)}||_F = \sqrt{12}$, $||V^{(2)}||_F = \sqrt{61}$, $||V^{(3)}||_F = \sqrt{351}$. In other words, with a perturbation budget of 7 for an adversarial attack scheme on X, it is possible to generate $X^{(1)}$, but it is impossible to generate $X^{(2)}$ or $X^{(3)}$. With a budget of 20, it is possible to generate each of the perturbed matrices.

Now, consider a probabilistic contamination model in the sense that with a certain probability, $1 - r$, a row of X stems from its original distribution, and with probability r, from some other distribution G. The first consequence is that the $V^{(k)}$ are no longer needed here, as there is no additive perturbation matrix. We assume for simplicity in this example that when a row of $X^{(k)}$ equals the respective row of X, it stems from the original distribution. We start with $X^{(2)}$. Only the first row differs from X. In this simplified example, it follows that for $r = 0.1$, the probability that only one row stems from another than the original distribution is $4 \cdot 0.1 \cdot 0.9^3$. However, whether realizing $X^{(2)}$ is possible and its likelihood also depends on the distribution G. If G is a $\mathcal{N}_3((6, -4, 3), \Sigma)$-distribution for some positive semi-definite matrix Σ, the realization of $X^{(2)}$ is certainly possible and its likelihood is given by the density of G at $(6, -4, 3)$. However, if G has zero density at $(6, -4, 3)$, realizing $X^{(2)}$ under this contamination scheme is impossible. A similar argumentation can be done for $X^{(3)}$. As for $X^{(1)}$, under our simplifying assumptions, having four rows realized from G is very improbable for a low r, but not impossible.

This example should emphasize that the main difference in the geometric distances used in adversarial attacks and the probabilistic distances that are encountered in contamination settings from Robust Statistics is that the former are deterministic in the sense that a certain adversarial contamination is either possible or impossible, while distributional contamination is more subtle and allows for a large variety of possible realized perturbations.

For the following definition, see [259, Sec. 4.2].

Def 3.1: Let $(\Omega, \mathcal{A})$ be a measurable space. Let $\mathcal{P} := \{P_{\boldsymbol{\theta}} \mid \boldsymbol{\theta} \in \Theta\}$ be a family of parametric distributions $P_{\boldsymbol{\theta}} \in \mathcal{P}_{\boldsymbol{\theta}}$ on $(\Omega, \mathcal{A})$, where $P_{\boldsymbol{\theta}_0}$ denotes the "ideal distribution". Let $\Theta \subset \mathbb{R}^p$ be a parameter space. A **contamination model** is given by the set $\mathcal{U}_*(\boldsymbol{\theta}_0) := \{U_*(\boldsymbol{\theta}_0, r) \mid r \in [0, \infty[\}$ of **contamination balls** $U_*(\boldsymbol{\theta}_0, r) = \{Q \in \mathcal{M}_1(\mathcal{A}) \mid d_*(P_{\boldsymbol{\theta}_0}, Q) \leq r\}$, where $\mathcal{M}_1(\mathcal{A})$ denotes the set of probability distributions on $\mathcal{A}$. One refers to r also as the "contamination radius".

One can consider the "ideal distribution" to be the distribution that one assumes for the underlying data, often idealized, e.g., Gaussian.

Example 3.2: The convex contamination model $\mathcal{U}_c(\boldsymbol{\theta}_0)$ considers a convex combination of distributions, leading to convex contamination balls of the form

$$U_c(\boldsymbol{\theta}_0, r) = \{(1 - r)_+ P_{\boldsymbol{\theta}_0} + \min(1, r) Q \mid Q \in \mathcal{M}_1(\mathcal{A})\}.$$

The convex contamination model is intuitive in the sense that with a probability of $\min(1, r)$, an instance in a data set, a gradient in neural network training, an action of an agent, or whatever the data consist of, is contaminated, i.e., in expectation, a $\min(1, r)$-fraction of the considered objects is not generated from the ideal distribution. In Ex. 3.1, convex contamination has been considered.

B. BREAKDOWN POINT

The breakdown point (BDP), roughly speaking, quantifies the amount of contamination that is necessary in order to achieve a breakdown of the estimator in the sense that the estimator may output unreasonable values. For a given data set, the so-called finite-sample BDP [78] is the relative fraction of instances that have to be contaminated in order to achieve such a breakdown. For regression, let the data set consist of instances $(\boldsymbol{X}_i, Y_i) \in \mathbb{R}^{p+1}$ and assume the model $\mathbb{E}[Y_i] = h(\boldsymbol{X}_i)\boldsymbol{\beta}$ for some unknown coefficient vector $\boldsymbol{\beta} \in \mathbb{R}^p$. The BDP is then defined as follows.

Def 3.2: Let $Z_n := \{(\boldsymbol{X}_1, Y_1), ..., (\boldsymbol{X}_n, Y_n)\}$ for instances $(\boldsymbol{X}_i, Y_i)$. The **case-wise finite-sample breakdown point** of an estimator $\hat{\boldsymbol{\beta}}$ for the regression parameter $\boldsymbol{\beta}$ is defined by

$$\varepsilon^*(\hat{\boldsymbol{\beta}}, Z_n) = \min \left\{ \frac{m}{n} \;\middle|\; \sup_{Z_n^m} (||\hat{\boldsymbol{\beta}}(Z_n^m)||) = \infty \right\}. \quad (1)$$

Here, the set Z_n^m denotes any sample that has exactly $(n-m)$ instances in common with Z_n. The coefficient $\hat{\boldsymbol{\beta}}(Z_n)$ denotes the estimated parameter on Z_n^m.

Note that the fraction given by the BDP is deterministic in the sense that, for example, in federated learning, one assumes that exactly m out of n gradients can be intercepted and manipulated by an attacker. In contrast, the convex contamination balls are stochastic in the sense that even if the contamination radius r is fixed, the number of contaminated objects follows an $B(n, \min(1,r))$-distribution.

We continue the Ex. 3.1 in order to illustrate the contaminations that are covered in BDP examinations.

Example 3.3: Consider the matrices X, $X^{(1)}$, $X^{(2)}$, and $X^{(3)}$ from Ex. 3.1. In contrast to the modeling approach with ideal and contaminating distributions as in Ex. 3.1, we inspect the situation where a breakdown point of some algorithm operating on X should be discussed. For comparison, we first consider adversarial attacks. Here, the question is whether some adversarial attack that can be crafted using the allowed budget can lead to a large deviation in the output of a trained model. Usually, one has a classification model and tries to find adversarial perturbations that cause the model to predict a different label for the perturbed input than for X (e.g., [111], [42]).

When dealing with poisoning attacks, one does not assume an already trained model as when considering adversarial attacks, but examines the impact of an adversarial perturbation w.r.t. some budget on the trained model itself, i.e., whether adversarial perturbations can distort the model during training.

For BDP inspections, the goal is similar as when considering poisoning attacks. Here, one is also interested in the impact of perturbations on the training process, but the contamination is injected differently. While poisoning attacks consider perturbations that are bounded by a geometric argument, e.g., the Frobenius norm, BDP discussions consider the fraction of perturbed observations only. In this sense, the perturbed matrices $X^{(k)}$ can appear in poisoning attack settings provided that the budget is sufficiently high. For BDP discussions however, the set Z_n^m is considered. In this example, the set Z_n^m consists of all $4 \times 3-$matrices for which exactly $(n-m)$ rows are identical with the respective rows of X. In this sense, the matrix $X^{(2)}$ lies in the set Z_n^m for all $m = 1, ..., 4$, while the matrix $X^{(3)}$ can only be considered for $m > 1$. The matrix $X^{(1)}$ would correspond only to $m = 4$, however, in nearly all settings, one would not allow for $m > n/2 = 2$, so one can assume that $X^{(1)}$ would not appear in BDP discussions.

The BDP concept has also been formulated, e.g., for classification (rotation of decision boundaries; [370]), ranking (order inversion; [325]), and clustering (dissolution of clusters; [136]).

In particular, in the context of high-dimensional data, [10] propose to consider the contamination of single cells. As one contaminated cell already contaminates the corresponding instance, in high-dimensional settings, one can easily contaminate each instance with a few outlying cells. One can nevertheless consider the relative fraction of contaminated cells as (cell-wise) BDP concept, see, e.g., [310], when analyzing cell-wise robust algorithms that are tailored to this setting.

C. INFLUENCE CURVE

Robust Statistics interprets estimators as statistical functionals, i.e., functionals which take a distribution as input. For example, the expected value of some distribution P can be denoted by the mean functional $T^{\text{mean}}(P) = \int \boldsymbol{x} P(d\boldsymbol{x})$. The influence curve goes back to [123]. The goal is to determine the local behavior of an estimator in a neighborhood around the ideal distribution by a suitable linearization of the underlying functional. Given such a linearization, i.e., a van-Mises expansion ([311]) of the statistical functional in the sense

$$T(Q) - T(P) = \int T'(P) d(Q-P)(\boldsymbol{x}) + \text{rem}$$

for some stochastic remainder term rem, the (Gâteaux-) deritative can be identified with the influence curve, i.e., $T'(P) = \text{IC}(\boldsymbol{x}, T, P)$.

Formally, the influence function is defined as follows (e.g., [259]).

Def 3.3: Let $\mathcal{Z}$ be a normed function space. Let further the parameter space Θ be a normed real vector space and let $T : \mathcal{Z} \to \Theta$ be a statistical functional. The **influence curve** of T at $\boldsymbol{x}$ for a distribution P on $\mathcal{Z}$ is given by

$$\text{IC}(\boldsymbol{x}, T, P) := \lim_{r \to 0} \left(\frac{T((1-r)P + r\delta_{\boldsymbol{x}}) - T(P)}{r} \right) = \partial_r \left[T((1-r)P + r\delta_{\boldsymbol{x}}) \right] \Big|_{=0}$$

for the Dirac measure $\delta_{\boldsymbol{x}}$ at $\boldsymbol{x}$.

The influence curve determines the infinitesimal impact of a single observation on the estimator. Robustness of the estimator in the sense of the influence curve requires that $|\text{IC}(\boldsymbol{x}, T, P)| < \infty$ for all $\boldsymbol{x}$. This property is called B-robustness.

D. ROBUST LOSSES AND AGGREGATION METHODS

Let a general M-estimator be given by

$$\hat{\boldsymbol{\theta}}^M = \text{argmin}_{\boldsymbol{\theta}} \left(\frac{1}{n} \sum\nolimits_{i=1}^{n} \rho(r_i(\boldsymbol{\theta})) \right)$$

for a loss function $\rho : \mathbb{R} \to \mathbb{R}$ and residuals $r_i(\boldsymbol{\theta})$. If ρ is differentiable, one can equivalently consider the corresponding Z-estimator

$$\text{zero}_{\boldsymbol{\theta}} \left(\frac{1}{n} \sum\nolimits_{i=1}^{n} \psi(r_i(\boldsymbol{\theta})) \right)$$

for $\psi = \rho'$. The influence function of an M-estimator is (e.g., [121])

$$\text{IC}(\boldsymbol{x}, \hat{\boldsymbol{\theta}}^M, P_{\boldsymbol{\theta}_0}) = -\frac{\psi(\boldsymbol{x})}{\mathbb{E}_{P_{\boldsymbol{\theta}_0}}[\psi'(\boldsymbol{X})]}. \tag{2}$$

Therefore, a robustification of an M-estimator can be done by bounding the derivative ψ of the loss function, which leads to "robust loss functions". A popular example is the Huber loss

$$\rho_H(r) = \begin{cases} r^2, & |r| \leq k \\ 2k|r| - k^2, & |r| \geq k \end{cases}$$

for a hyperparameter k. One can interpret location M-estimators as weighted means of the form (e.g., [213])

$$\sum_{i=1}^{n} w(\boldsymbol{X}_i - \hat{\boldsymbol{\theta}})(\boldsymbol{X}_i - \hat{\boldsymbol{\theta}}) = 0, \quad \hat{\boldsymbol{\theta}} = \frac{\sum_i w_i \boldsymbol{X}_i}{\sum_i w_i}, \quad w_i = w(\boldsymbol{X}_i - \hat{\boldsymbol{\theta}}).$$

In the case the Huber loss, the weight function w is given by

$$w_H(r) = \min\left(1, \frac{k}{|r|}\right).$$

A disadvantage of the Huber loss is that the loss function is still unbounded, which makes Huberized M-estimators vulnerable against large outliers or heavy-tailed distributions. In order to cope with such situations, one uses loss functions which are bounded, implying that their derivatives ψ tend to zero again in the sense $\lim_{|r|\to\infty}(\psi(r)) = 0$. Therefore, such derivatives are sometimes called "redescenders". A popular loss function of this type is the Tukey-biweight loss, given by

$$\rho_T(r) = \begin{cases} 1 - \left(1 - \left(\frac{r}{k}\right)^2\right)^3, & |r| \leq k \\ 1, & |r| \geq k \end{cases}.$$

Further losses with redescending derivative are the Welsch loss

$$\rho_W(r) = 1 - \exp\left(-\frac{1}{2}\left(\frac{r}{k}\right)^2\right),$$

the Geman-McClure loss

$$\rho_{GM}(r) = \frac{r^2}{r^2 + k^2},$$

and the Cauchy loss

$$\rho_C(r) = \frac{k^2}{2} \ln\left(1 + \left(\frac{r}{k}\right)^2\right).$$

Another technique, which still allows for using standard loss functions such as the squared loss, is to robustify the aggregation of the losses corresponding to the individual instances. This is done by trimming, leading to approaches such as the least median of squares [262]

$$\text{argmin}_{\boldsymbol{\theta}}(\text{med}(r_i(\boldsymbol{\theta}))).$$

Due to a slow convergence rate, [262] proposed the least trimmed squares (LTS) estimator

$$\text{argmin}_{\boldsymbol{\theta}}\left(\sum_{i=1}^{h}(r^2(\boldsymbol{\theta}))_{i:n}\right),$$

where $z_{1:n}$ denotes the smallest element of a vector $\boldsymbol{z} \in \mathbb{R}^n$, $z_{2:n}$ the second smallest element and so forth. In other words, the LTS estimator intends to minimize the sum of squares for the $h \leq n$ observations with the smallest squared residuals (the set of these instances is sometimes called "clean subset"). Due to the computational complexity of LTS, [264], [265] proposed an iterative algorithm that starts with an initial subset $I_h^{(0)}$ of size h so that the parameters of the model are computed solely using the instances on $I_h^{(0)}$. Then, the residuals for all instances are computed, leading to the next iterate $I_h^{(1)}$ that consists of the h instances with the smallest residuals. Due to only attaining a local minimum, this algorithm is repeated for several initial sets, so that the final h-set with the smallest sum of residuals (over this h-set) is taken. This technique has been extended to high-dimensional models by the Sparse LTS (SLTS) method [7] where Lasso models are computed in each iteration.

In regression, one may have to estimate both the regression parameter $\hat{\boldsymbol{\beta}}$ and a scale $\hat{\sigma}$. This can be done by first estimating $\hat{\sigma}$ and by solving

$$\text{argmin}_{\boldsymbol{\beta}}\left(\frac{1}{n}\sum_i \rho\left(\frac{r_i(\boldsymbol{\beta})}{\hat{\sigma}}\right)\right),$$

for a bounded loss function ρ. The idea of MM-estimators is to first compute a consistent and highly robust estimator $\hat{\boldsymbol{\beta}}^{(0)}$, to compute a robust scale estimator $\hat{\sigma}$ and to find a solution of the problem above, allowing for both robustness and high efficiency.

E. OTHER NOTIONS OF ROBUSTNESS IN AUTONOMOUS DRIVING

The term "robustness" is often used in the AI literature, including that on autonomous driving, in a dictionary-sense such as robustness against error propagation by the simplification of computation steps or against hyperparameter settings of a certain algorithm. Robustness can also be understood as a better accessibility of model parameters (e.g., [81]).

The closest understanding of robustness to that from Robust Statistics is the consideration of challenging environment conditions such as GPS in the presence of tunnels and canyons [219], sensor fusion in "hostile environments" [339] or rain [360], or in general the gap between a simulation and the real world, e.g., [8], [11]. Sensors such as LiDAR or Radar that can cope with varying lightning or weather conditions are also called "robust" [89].

In deep learning in general, the term "robustness" is often understood as **adversarial robustness** (e.g., [156]; see, e.g., [24], [266] for details on adversarial robustness), which is not the core understanding of robustness in the sense of Robust Statistics, because the perturbation occurs after model training, while Robust Statistics considers the effect of contamination onto the estimator, i.e., the contamination appears before training and therefore potentially affects the trained model. [104] point out that there are different understandings of the term "robustness" and focus themselves on the classical robustness in terms of the breakdown point. Moreover, they correctly emphasize that robustness in the sense of the BDP does not guarantee adversarial robustness.

At least two measures for adversarial robustness have been proposed in the literature, the error-rate-based measure [212] where adversarial samples are generated given a certain perturbation radius and the relative number of errors is investigated, or the radius-based measure [296] where one searches

for the minimum perturbation radius in order to generate a misclassification. In the adversarial setting, one uses the term "certified robustness" which indicates some guarantee that an adversary does not have success provided that the perturbation norm is smaller than some threshold. The counterpart from Robust Statistics is the property of a non-zero BDP in order to guarantee global robustness here.

Sometimes even the convex contamination setting is interpreted as adversarial setting [371]. [382] use the terminology "robust loss" for a worst-case loss in adversarial training. [338] call the property that machine learning models perform well even in the presence of adversarial attacks as "robust accuracy" or "robust generalization". [112] define "over-robustness" (for Graphical Neural Networks (GNNs)) as unwanted robustness in the sense that even the semantic context has changed due to the perturbations, the robust classifier does not react, which they call "robust beyond the point of semantic change". The term "trigger/backdoor robustness" has been coined in [118] who consider backdoor and poisoning attacks.

[135] speak of "common corruptions" of images such as blur, Gaussian noise or due to certain weather conditions such as fog. They propose not to only consider the worst-case situation when assessing robustness as in adversarial attacks (and, notably, also in BDP computations) but to also take these common corruptions into account. They introduce the term "corruption robustness" which does not refer to the minimum probability that the classifier predicts the correct class over a perturbation ball as in adversarial robustness but which refers to the expectation over a set of corruptions. Note that this idea is similar to the expected finite-sample BDP from [268] where one abstains from considering the worst-case contamination in the context of heavy-tailed distributions.

F. RANSAC

A popular algorithm that entered autonomous driving application is RANSAC (random sample consensus), going back to [92]. The idea of RANSAC is to iteratively identify the worst points (usually time points) and to remove them from the data. More precisely, RANSAC samples $m < n$ instances, computes a model $f_{\hat{\theta}}$ and determines the consensus set, which is given by the instances for which the loss is smaller than some threshold. If the size of this set is larger than h for some h, one uses this set to re-compute the model, otherwise, one samples another random subset of size m and repeats the procedure. At the end, the largest consensus set (which one may again interpret as "clean" subset) observed is reported. The elements of this consensus set are interpreted as inliers here. This procedure can be interpreted as a brute-force counterpart of the iterative algorithm for the computation of the LTS.

For example, [349] use the RANSAC algorithm in order to address ego-motion estimation, segmentation, and moving object detection. They point out that RANSAC is tailored to environments with rapid changes. [178] apply RANSAC for robust pose estimation of vehicles.

[291] point out that on data corresponding to rotation search or point cloud registration, one can even have more than 95% outliers, see also [239], and work with up to 99% outliers in their experiments. The reason is that for two point clouds $\mathcal{P} = (\boldsymbol{P}_i)_{i=1}^n$, $\mathcal{P}^* = (\boldsymbol{P}_i^*)_{i=1}^n$, $\boldsymbol{P}_i, \boldsymbol{P}_i^* \in \mathbb{R}^3$, mismatched keypoints or localization errors can result in a lot of false correspondences $(\boldsymbol{P}_i, \boldsymbol{P}_i^*)$ [291], [178]. From the perspective of Robust Statistics, such a high contamination radius is uncommon, and most concepts can at most deal with contamination radii of 0.5 because the BDP of an equivariant estimator cannot exceed 0.5 asymptotically [71]. There are however at least two cases where the number of outlying instances is allowed to be higher. First, when aggregating models, e.g., [326] proposed a trimmed Stability Selection where only the models corresponding to the smallest out-of-sample losses are considered for aggregation, theoretically allowing for a higher rate of outliers in the data set than 0.5 because resampling can result in sufficiently clean training batches (where the outlier ratio is at most 0.5). On the other hand, an instance is considered to be outlying if at least one cell is contaminated [10]. In such situations, cell-wise robust algorithms provide an alternative to classical robust algorithms as they can deal with the situation that each instance is contaminated, provided that the cell-wise contamination rate is lower than their cell-wise BDP. However, this requires sufficiently many predictor variables, e.g., in the setting of point cloud registration, the data set only consists of the pair-wise point correspondences, making the notion cell-wise robustness obsolete.

RANSAC has disadvantages, such as the long computation time, the dependence on the minimum number of instances that is required for defining a model [254], the problem to apply it to data with only a few samples due to sparse measurements or many dropouts [210], the increased complexity for large outlier fractions [283], the sensitivity to the outlier threshold [210], its non-deterministic nature (e.g., [239], [280]), and the difficulty to apply it to high-dimensional problems [280]. RANSAC clearly depends on the error threshold, which leads to the problem that, in contrast to trimming approaches such as LTS where the trimming rate (i.e., α such that $h = \lceil (1-\alpha)n \rceil$) is fixed, defining a threshold does not provide ex ante information to how many non-trimmed instances it corresponds. According to [341], RANSAC can deal with 80% outliers but becomes very expensive due to re-adaptations when the outlier rate is higher than 90%.

[210] experimentally compare different M-estimators with the squared, absolute, Huber, Cauchy, Geman-McClure, the dynamically scaled covariance loss [4] and a clipped squared loss on a dataset for visual localization, with the result that the Geman-McClure loss, optionally combined with clipping in the sense that the last iterations are done w.r.t. the clipped squared loss as loss function instead of the Geman-McClure loss, lead to the best results in the presence of large contamination radii. RANSAC cannot be applied due to the large contamination radius, resulting in too few correct correspondences in each images. [225] showed that RANSAC

in combination with robust base estimators such as LTS or LMS performs better in the presence of contamination than RANSAC with standard least squares. The methods were evaluated by the number of inliers they identified on simulated data where this number is known. [182] point out that the iterative closest point algorithm (ICP), which is the standard tool for point cloud registration, heavily depends on the initializations of the transformations and that it cannot deal with cross-source point clouds, for example, from multiview stereo. They combine RANSAC with the Tukey biweight in order to overcome the problem of a very high required number of trials of RANSAC.

[140] propose VODRAC (voting-based double-point random sampling with compatibility weighting). The idea is to overcome to computational complexity of RANSAC by using the pairwise compatibility constraint. That is, for the model $\tilde{\boldsymbol{p}}_i^* = R\tilde{\boldsymbol{p}}_i + \vec{\boldsymbol{t}} + \boldsymbol{\epsilon}_i$ for $\tilde{\boldsymbol{p}}_i$ and $\tilde{\boldsymbol{p}}_i^*$ from point clouds $\mathcal{P}$, $\mathcal{P}^*$, respectively, a rotation operator $R \in \mathrm{SO}(3)$ and a translation $\vec{\boldsymbol{t}} \in \mathbb{R}^3$, the constraint is

$$r_{ij} := ||\tilde{\boldsymbol{p}}_i^* - \tilde{\boldsymbol{p}}_j^*|| - ||\tilde{\boldsymbol{p}}_i - \tilde{\boldsymbol{p}}_j|| \leq 2\eta$$

for the inlier threshold η. This norm difference equals $2||\boldsymbol{\epsilon}_i - \boldsymbol{\epsilon}_j||$ under the model above. This allows for checking whether two correspondences are compatible, then, one can check whether a third correspondence is compatible with each of these two correspondences and so forth, facilitating the search of the inlier set. This technique is referred to a double-point random sampling. In addition, they aim at putting more weight onto clear inliers, i.e., for which r_{ij} is small, by invoking Tukey's biweight loss function, leading to the weights

$$w_{ij} = \begin{cases} \left(1 - \frac{r_{ij}^2}{(2\eta)^2}\right)^2, & r_{ij}^2 \leq (2\eta)^2 \\ 0, & r_{ij}^2 > (2\eta)^2 \end{cases},$$

which allows a sorting of the correspondence set in the sense that the minimal subset is formed by the correspondences with the highest weights.

IV. APPLICATIONS IN AUTONOMOUS DRIVING: PERCEPTION

This section collects approaches based on Robust Statistics in perception tasks for autonomous driving. In each subsection, we address one of the sub-tasks that we already listed in Fig. 1. Robust perception refers to strategies that allow for corrupted data, such as outliers in camera or LiDAR data, that may result from challenging weather conditions, light reflections, occlusions, or just measurement errors. The extraction of realiable state information from those data is vital in order to suitably predict the maneuvers of other traffic participants and to plan own maneuvers.

A. SLAM

[141] propose to use the Cauchy loss for a robust graph-based SLAM for the model

$$\boldsymbol{z}_{ij} = h_{ij}(\boldsymbol{p}_i, \boldsymbol{p}_j) + \boldsymbol{\epsilon}_{ij}$$

for a non-linear measurement function h_{ij}, positions $\boldsymbol{p}_i$, errors $\boldsymbol{\epsilon}_{ij}$ and measurements $\boldsymbol{z}_{ij}$ from $\boldsymbol{p}_i$ to $\boldsymbol{p}_j$. The goal is to estimate the true locations $\boldsymbol{p}_i$, so the residuals that enter the Cauchy loss function are

$$r_{ij} = ||\Sigma_{ij}^{1/2}(\boldsymbol{z}_{ij} - h_{ij}(\boldsymbol{p}_i, \boldsymbol{p}_j))||_2$$

for the covariance matrix Σ_{ij}^{-1} of the ideal model $\boldsymbol{\epsilon}_{ij} \sim \mathcal{N}(\boldsymbol{0}, \Sigma_{ij}^{-1})$. The objective is then

$$\mathrm{argmin}_{(\boldsymbol{p}_1,\ldots,\boldsymbol{p}_n)} \left(\sum\nolimits_{(i,j)\in E} \rho_C(r_{ij}) \right)$$

for the edge set E of the corresponding SLAM graph.

[4] also consider a graph-based approach of the SLAM model which aims to minimize

$$\sum\nolimits_t ||h_{t,t+1}(\boldsymbol{p}_t, \boldsymbol{p}_{t+1}) - \boldsymbol{z}_{t,t+1}||_{\Sigma_t}^2 + \sum\nolimits_t \sum\nolimits_{t'} ||f(\boldsymbol{p}_t, \boldsymbol{p}_{t'}) - \boldsymbol{z}_{t,t'}||_{\Lambda_{t,t'}}^2$$

where the indices t and t' correspond to time steps. The covariance matrices of the odometry and sensor measurements are given by Σ_t and $\Lambda_{t,t'}$, respectively. The goal is to find the positions $\boldsymbol{p}_t$ that minimize the loss. They propose the dynamically scaled covariance loss,

$$\begin{aligned} &\sum\nolimits_t ||h_{t,t+1}(\boldsymbol{p}_t, \boldsymbol{p}_{t+1}) - \boldsymbol{z}_{t,t+1}||_{\Sigma_t}^2 + \\ &\sum\nolimits_t \sum\nolimits_{t'} ||\Psi(\zeta_{t,t'}) h_{t,t'}(\boldsymbol{p}_t, \boldsymbol{p}_{t'}) - \boldsymbol{z}_{t,t'}||_{\Lambda_{t,t'}}^2 + \\ &\sum\nolimits_t \sum\nolimits_{t'} ||1 - \zeta_{t,t'}||_{\Xi_{t,t'}}^2, \end{aligned}$$

where the $\zeta_{t,t'} \in [0,1]$ are switching variables, $\Psi : [0,1] \to [0,1]$ is a scaling function and where $\Xi_{t,t'}$ is a switching prior. This loss is minimizes w.r.t. both the $\boldsymbol{p}_t$ and the $\zeta_{t,t'}$. They show that the solution is given by

$$\zeta_{t,t'} = \min\left(1, \frac{2\Xi_{t,t'}^{-1}}{\Xi_{t,t'}^{-1}} + ||h_{t,t'}(\boldsymbol{p}_t, \boldsymbol{p}_{t'}) - \boldsymbol{z}_{t,t'}||_{\Lambda_{t,t'}}^2\right).$$

[3], [210] show that, inserting the unconstrained solution for the $\zeta_{t,t+1}$ (so that they are not upper bounded by 1) for dynamically scaled covariance, into the loss function, one replicates the Geman-McClure loss function, up to a constant factor. [210] identify the dynamically scaled covariance loss therefore with a variant of the Huber loss where the squared loss is used for small residuals, and the Geman-McClure loss for large residuals.

[3] propose a Bayesian approach for estimating the posterior of the state (position) variables, so that robust loss functions can be implicitly encoded via corresponding distributions such as corrupted Gaussian in a mixture approach. They test their procedure using real data from Google StreetView maps from which the necessary poses and 3D points are extracted.

[303] use the truncated least squares loss for location estimation, i.e.,

$$\sum\nolimits_i \sum\nolimits_j \rho(r_{ij}^2), \quad \rho(r) = \begin{cases} r^2, & |r| \leq k \\ k^2, & |r| \geq k \end{cases},$$

for the geodesic distances r_{ij}^2 between the average poses and the measured poses.

[202] consider multimodal motion prediction and propose a loss function composed by several losses, one is a regression loss w.r.t. the coordinate offsets for which they use the Huber loss. The goal is to predict trajectories until a given horizon.

Many SLAM approaches consider GPS or GNSS data, where one usually has the (pseudo)range as response variable.

[96] compare several robust regression by applying non-robust and robust estimation methods for positioning estimation in challenging areas such as urban canyons or city centers. Their model is given by

$$Y_i = \boldsymbol{X}_i\boldsymbol{\beta} + \epsilon_i \tag{3}$$

where the Y_i are the differences between measured and predicted pseudoranges, the $\boldsymbol{X}_i$ are the geometry matrices and $\boldsymbol{\beta}$ is a vector consisting of the receiver coordinates and the clock offset of the receiver and the satellite, scaled with the speed of light. As for the robust methods, LTS and M-estimation with the Huber loss and the IGGIII weight function

$$w_{\text{IGGIII}}(r) = \begin{cases} 1, & |r| \le k_1 \\ \frac{k_1}{|r|}\left(\frac{k_2-|r|}{k_2-k_1}\right)^2, & k_1 \le |r| \le k_2 \\ 0, & |r| \ge k_2 \end{cases},$$

respectively, are applied.

Another comparison has been made in [173] who compare several robust regression and outlier detection methods, including LTS, LMS, robust M-estimators, S-estimators, and MM-estimators, on simulated GPS data where the response variable is the pseudorange. The goal of [173] was to study how many outliers were correctly detected by the individual methods. They point out that the robust methods are time-consuming and may hinder real-time performance.

[9] apply robust regression methods on GNSS data, where they consider the linear model Eq. 3 where $\boldsymbol{X}_i$ at least contains information about the satellite ID, the epoch and the elevation and where Y_i are the pre-fit pseudo-ranges. They apply LTS, LMS and a forward search, where one starts searching for a clean subset of size p and increases this number iteratively. They achieve real-time capability and propose to not analyze large chunks of data at once but to use a sliding-window approach. [6] consider MM-regression.

[12] propose to apply the Huber M-estimator for GPS position estimation. The underlying model is given by

$$Y_i = \boldsymbol{X}_i^{(d)} + \boldsymbol{X}_i^{(b)} + \epsilon_i,$$

where Y is a pseudorange measurement, $X^{(d)}$ is the geometric distance from satellite to receiver and $X^{(b)}$ is a receiver clock offset. In the linear model Eq. 3, the parameter vector $\boldsymbol{\beta}$ contains the incremental corrections to the unknown variables (receiver coordinates and clock offsets). Due to linear relationships of the residuals and measurement errors, they compute the redundancy matrix which is used to modify the residuals. See also [64] for an application of robust M-estimators for GNSS in urban scenarios, where a three-satellite constellation is considered, which is reflected by three clock offsets in the features. [217], [218], [216] consider the regression problem from [12] with the tropospheric and ionospheric corrections as additional features in the model above and also use S- and MM-estimators. [40] also integrate ionospheric and tropospheric corrections and replace the WLS estimation by an estimation based on the Huber loss. [361] additionally include multi-path delays and ionospheric and tropospheric corrections in the pseudo-range model. Using real-world data from open-sky, semi-urban and dense-urban environments, they apply different robust loss functions, including the Huber, Tukey, Cauchy, Geman-McClure and Welsch loss.

[114] propose to adapt the threshold of Tukey's biweight loss for GNSS position estimation. This is done in dependence of the detected fraction of multipaths in the data when applying a CNN. The higher this fraction, the lower the threshold, which is chosen in order to maintain a given efficiency or BDP. Alternatively, they propose a robust M-estimator, which is computed via IRWLS.

[358] aim at discarding pseudorange and Doppler measurements in GNSS. They point out that Doppler measurements are also affected by reflections from buildings or trees, although to a smaller extent than pseudorange measurements. Based on the NFA (number of false alarms) criterion, i.e.,

$$\text{NFA}(D) = \eta \frac{1}{\Gamma(|D|/2)} \int_0^{\delta_D^2/(2\sigma^2)} e^{-t} t^{|D|/2-1} dt,$$

for the set D of observations, a normalization constant η, the variance σ^2 of the underlying assumed normal distribution of the measurement noise, and the sum of squares δ_D^2 of the standardized residuals, they propose an iterative algorithm in order to find a "clean subset" of the data that minimizes this criterion.

[320] propose a cross-view localization based on both satellite and ground views. Given feature maps extracted by a CNN from the satellite and ground-view images, the residuals $\boldsymbol{r}_i$ between the components of these feature maps are computed. Then, the individual points are weighted according to weights that are proportional to the derivative of some robust loss function so that points with large residuals are downweighted.

[237] consider robust range estimation and propose to use a $\tanh$-type robust loss function of the form

$$\rho(Y_i, \hat{Y}_i) = \frac{1}{\beta} \tanh\left(\frac{\beta}{2}\frac{(Y_i - \hat{Y}_i)^2}{\sigma^2}\right),$$

where the Y_i are the measured ranges, $\hat{Y}_i$ their predicted counterparts, and where $\sigma > 0$ is a scale parameter. The parameter β is estimated by LMS, σ by the MAD. They also propose a robust Bayesian algorithm, which is initiated by the weights,

$$w(Y_i, \hat{Y}_i) = \frac{1}{\sigma^2}\text{sech}^2\left(\frac{\beta(Y_i - \hat{Y}_i)^2}{2\sigma^2}\right),$$

computed from the M-estimation.

[323] propose the GNSS measurement model

$$Y_{ij} = ||\boldsymbol{X}_i - \boldsymbol{a}_j|| + c(\delta_i^s - \delta_j) + I_{i,j} + T_{i,j} + b_{i,j} + \epsilon_{i,j}$$

for the pseudorange between vehicle i and satellite n, where $\boldsymbol{a}_j$ is the position of satellite j, where δ_i^s and δ_j are the clock offset of vehicle i and satellite j to the satellite system s, respectively, and where $\epsilon_{i,j}$ is the measurement noise. Furthermore, c denotes the speed of light and $I_{i,j}$ and $T_{i,j}$ denote measurement errors that are induced by the ionosphere and the troposphere, respectively. Lastly, $b_{i,j}$ are latent variables for modeling unknown measurement biases. As for vehicle to vehicle measurements, they use the model

$$Y_{ij} = h(\boldsymbol{X}_i, \boldsymbol{X}_j) + b_{ij} + \epsilon_{ij},$$

for latent variables b_{ij}, measurement noise ϵ_{ij} and the states $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ of vehicle i and vehicle j, respectively. Assuming that the measurement noise is Gaussian, any contamination is modelled by the latent variables. To this end, Gaussian-Gamma prior distributions are assumed, and the joint distribution of the states and latent variables are approximated via variational inference. They apply their method on a real-world data set with three vehicles. [40] consider the same GNSS model, but without latent variables. In a differential GNSS approach, the differences

$$\boldsymbol{\rho}^b = h(\boldsymbol{X}) - h(\boldsymbol{X}^b) + \boldsymbol{\epsilon}^b$$

are modelled, where $\boldsymbol{X}^b$ denotes the position of the base station. First-order linearization leads to

$$\boldsymbol{\rho}^b \approx h(\boldsymbol{X}_0) + H_r\boldsymbol{\delta} - h(\boldsymbol{X}^b) + \boldsymbol{\epsilon}^b,$$

for $\boldsymbol{\delta} = \boldsymbol{X}_0 - \boldsymbol{X}$ and geometry matrix H_r. The residuals are given by $\boldsymbol{r}^b \approx H_r\boldsymbol{\delta} + \boldsymbol{\epsilon}^b$ and modelled by a Gaussian distribution. In the collaborative localization setting, the measurements of all individual vehicles are concatenated. Denoting the concatenated counterparts of the quantities above by $\tilde{\boldsymbol{r}}$, $\tilde{H}_r$ and $\tilde{\boldsymbol{\delta}}$, the Gaussian assumption allows for a WLS formulation of the form

$$\hat{\tilde{\boldsymbol{\delta}}} = \text{argmin}_{\tilde{\boldsymbol{\delta}}}\left((\tilde{\boldsymbol{r}} - \tilde{H}_r\tilde{\boldsymbol{\delta}})^T\tilde{W}(\tilde{\boldsymbol{r}} - \tilde{H}_r\tilde{\boldsymbol{\delta}})\right),$$

where the weight matrix is the inverse joint covariance matrix.

[107] propose a baro-radar odometry approach based on barometry and radar and use robust loss functions for the barometry and Doppler residuals. Radar data are also considered in [192] who propose to use a truncated least squares loss function.

1) Visual odometry/ego-motion estimation

Many approaches intend to find a linear transformation that relates the 2D images collected from the camera and 2D reference images.

[62], [63] consider dense visual tracking under large illumination changes. Given a stereo camera pair, making n intensity measurements each, these observations are stored into two sets I and I'. Let $\mathcal{I} = (I, I')^T$ be the current view pair, let $\mathcal{I}^*$ be the reference view pair, and let $\mathcal{P}^* = \{\boldsymbol{p}, \boldsymbol{p}'\}$ be a set of stereo image correspondences (the pixel locations) from a pair of reference templates from the set $\mathcal{P}_n^* = \{\{\boldsymbol{p}^*, (\boldsymbol{p}')^*\}_1, ..., \{\boldsymbol{p}^*, (\boldsymbol{p}')^*\}_n\}$. Let f be the motion model, which is quadrifocal warping in [62], represented by a transformation $T \in \text{SE}(3)$. If $\bar{T}$ is the true transformation, and if $\hat{T}$ is the estimated transformation until time step $(t-1)$, the tracking problem then amounts to estimating the incremental transformation $T(\boldsymbol{\xi})$ at the current time step t, under the assumption that there exists $\boldsymbol{\xi}_0$ such that $T(\boldsymbol{\xi}_0)\hat{T} = \bar{T}$. Then, the standard criterion based on the least-squares cost is

$$\sum\nolimits_{\mathcal{P}^* \in \mathcal{P}_n^*}(I(f(\mathcal{P}^*, T(\boldsymbol{\xi})\hat{T}) - \mathcal{I}^*(\mathcal{P}^*)))^2,$$

where the quadratic loss is replaced by the Huber loss in [62], [63]. Experiments on real-world data show that their approach allows for real-time performance. They also make suggestions for further improvement.

[219] propose a hybrid approach between model-based optimization, where the error between the current model and the transformed current image is minimized, and VO, which minimizes the distance between the previous and current transformed image. More precisely, for model-based tracking, they assume the relation

$$i^*(\mathcal{P}^*) = \alpha i_t(f(\mathcal{P}^*, \bar{T}) - \beta$$

between the reference image intensities, i^*, and the current image intensities i_t at time t, leading to the Huberized objective

$$\sum\nolimits_{\mathcal{P}^* \in \mathcal{P}_n^*}\rho_H(\alpha i_t(f(\mathcal{P}^*, T(\boldsymbol{\xi})\hat{T}) - \beta - i^*(\mathcal{P}^*))).$$

In the VO approach, they consider augmented reference images that include the warped image i_{t-1}^f from the previous time step, where

$$i_{t-1}^f(\mathcal{P}^*) = i_{t-1}(f(\mathcal{P}^*, T_{t-1})),$$

leading to a similar objective as above but where $i^*(\mathcal{P}^*)$ is replaced by $i_{t-1}^f(\mathcal{P}^*)$. As the model-based approach suffers from illumination changes (apart from the fact that it requires an à priori model, which may be very difficult to obtain, as pointed out in [63]), and the VO approach is prone to drift due to the accumulation of errors during feature extraction and matching [63], [219] combine both approaches, the robust loss functions corresponding to both approaches are stacked so that a joint optimization is performed. [219] achieve near real-time performance on real-world data with a stereo camera pair.

[229] propose a Huberized approach in model-based visual tracking by downweighting the contribution of all pixels whose photometric error is higher than some iteratively decreasing threshold. The standard average photometric error is given by

$$C_r(\boldsymbol{p}, \boldsymbol{d}) = \frac{1}{|\mathcal{J}^*|}\sum\nolimits_{I \in \mathcal{J}^*}||\boldsymbol{r}_{ph}(\boldsymbol{I}, \boldsymbol{p}, \boldsymbol{d})||_1,$$

$$\boldsymbol{r}_{ph}(\boldsymbol{I},\boldsymbol{p},\boldsymbol{d}) = \boldsymbol{I}^*(\boldsymbol{p}) - \boldsymbol{I}(\pi(KT\pi^{-1}(\boldsymbol{p},\boldsymbol{d}))$$

for inverse depth $\boldsymbol{d}$, a set $\mathcal{J}^*$ of indices of reference images from the set $\mathcal{I}^*$, $\boldsymbol{I}^* \in \mathcal{I}^*$, pixel $\boldsymbol{p} = (x,y)^T$, $T \in \mathrm{SE}(3)$, a camera-intrinsic transformation matrix K, and back-projection $\pi^{-1}(\boldsymbol{p},\boldsymbol{d})$ of the inverse depth value to a 3D point. The robustification now invokes the Huber norm

$$||\boldsymbol{r}||_\epsilon = \begin{cases} \frac{||\boldsymbol{r}||_2^2}{2\epsilon}, & ||\boldsymbol{r}||_2 \leq \epsilon \\ ||\boldsymbol{r}||_1 - \frac{\epsilon}{2}, & ||\boldsymbol{r}||_2 \geq \epsilon \end{cases},$$

which enters the energy functional

$$E_\aleph = \int w(\boldsymbol{u})||\boldsymbol{\nabla}\aleph(\boldsymbol{u})||_\epsilon + \lambda C(\boldsymbol{u},\aleph(\boldsymbol{u}))d\boldsymbol{u}$$

for the map $\aleph$ that assigns a depth value to a pixel, and a pixel weight function w. Similar approaches based on robust loss functions can be found in [238], where the Huber function is directly applied to the $\boldsymbol{r}_{ph}$, and [110], [172], who consider the IRWLS formulation of the minimization problem w.r.t. the photometric error, where they use robust weight functions such as the Huber of Tukey weight function. Experiments on real-world data sets confirm real-time capability. [171], [169] consider the photometric residuals $\boldsymbol{r}_{ph}$ and formulate the MAP estimation problem

$$\mathrm{argmax}_{\boldsymbol{\xi}}(P(T(\boldsymbol{\xi})|(\boldsymbol{r}_{ph}(\boldsymbol{I},\boldsymbol{u},\boldsymbol{d}))_{\boldsymbol{I}\in\mathcal{J}^*})),$$

searching for the transformation $T(\boldsymbol{\xi})$ that maximizes the posterior probability of the residuals. Here, they allow for heavy-tailed distributions such a t-distribution. [171] apply their approach on data from an autonomous flight experiment and achieve real-time performance.

[343] propose to minimize the geometric projection error instead of the photometric error due to a higher resistance against illumination changes. The idea is to find a distance transform map D_c that computes the Euclidean distance to the closest edge for each pixel. For an edge pixel $\boldsymbol{e}_i$ from the current frame I_t, it should therefore hold that $D_c(\boldsymbol{e}_i) = 0$. Let the reprojection residual for an edge pixel $\boldsymbol{e}_i^*$ from a reference image I^* be

$$r(\boldsymbol{e}_i^*) = D_c(\hat{\boldsymbol{e}}_i)$$

for the reprojection position $\hat{\boldsymbol{e}}_i$ computed by the underlying rotation and translation model. The objective is

$$\sum\nolimits_{\boldsymbol{e}_i^*\in\mathcal{E}^*} ||\boldsymbol{r}(\boldsymbol{e}_i^*)||_\epsilon$$

for the Huber norm $||\cdot||_\epsilon$ and the set $\mathcal{E}^*$ of all edges in I^*. Real-time performance has been shown on real-world data sets.

[17] consider dense VO [169], which does not only use matched features as sparse VO does, i.e., dense VO uses all pixels, resulting usually in a higher precision but at higher computational costs. They point out that using a t-distribution for both geometric and photometric errors ignores the physical process, resulting in photometric errors not being well-represented by such a noise model. Therefore, they propose to use a t-distribution for photometric errors but a probabilistic sensor noise model for geometric errors (which in turn is not suitable for photometric errors), and estimate the transformation between the 3D camera coordinates and 2D image points.

[381] argue that photo bundle adjustment (PBA), which estimates scene geometry and camera motion in VO, is usually done by minimizing the photometric error. Motivated by works such as [169], they point out that PBA must be robustified against outliers that may arise due to widely separated active key frames so that the photo-consistency assumption may be violated by occlusions and reflections. In [381], their PBA error function for the total photometric error has the form

$$\sum\sum\sum\sum w(r_i)r_i^2(\boldsymbol{\xi})$$

for the parameters $\boldsymbol{\xi}$, the squared residuals r_i^2 and weights $w(r_i)$, where the quadruple sum goes over all pixels in all points corresponding to the active keyframes. The problem is that the usually used Levenberg-Marquardt algorithm in order to optimize this objective picks keyframes according to photometric consistency, so that frames with occlusions or reflections are prone to be ignored here. Although [381] consider sparse VO, they conclude that a t-distribution is also suitable for the photometric errors as in dense VO considered in [17]. They first derive that the approach based on the t-distribution is also suitable here, and also make experiments with the Huber weights

$$w(r_i) = \begin{cases} \sigma^{-2}, & |r_i| < k \\ k\sigma^{-2}|r_i|^{-1}, & |r_i| \geq k \end{cases},$$

where σ^2 is the variance of the ideal Gaussian distribution of the photometric errors. Experiments reveal that the t-distribution leads to even better performance because the weights drop even faster at the tails. In their experiments, they also flag points as outliers if the number of outlying pixels (flagged as such if the photometric error exceeds the $95\%-$quantile of the error in the respective keyframe) exceeds some threshold and delete them from the set of observations. The Huber loss is also used in [94] and [221]. Experiments on KITTI and other data sets confirm real-time performance.

[155] consider the problem of camera ego-motion estimation and propose a robust ego-motion estimation procedure. They argue that the noise in real-time flow data is often non-Gaussian and that violations of the scene-rigidity assumption due to objects moving independently result in outliers. The underlying model is

$$\boldsymbol{u}(\boldsymbol{p}_i) = \boldsymbol{\delta}(\boldsymbol{p}_i)A(\boldsymbol{p}_i)\boldsymbol{v}_t + B(\boldsymbol{p}_i)\boldsymbol{v}_r$$

for the optical flow $\boldsymbol{u}(\boldsymbol{p}_i)$ at image position $\boldsymbol{p}_i \in \mathbb{R}^2$, the translational velocity $\boldsymbol{v}_t \in \mathbb{R}^3$, the rotational velocity $\boldsymbol{v}_r \in \mathbb{R}^3$, the inverse $\boldsymbol{\delta}$ of the scene depth and linear transformations A and B. Motivated by [363], who already proposed a robust ego-motion estimation procedure based on IRWLS, they write the problem as a regression problem as [363], i.e.,

$$\min_{\boldsymbol{v}_r,\boldsymbol{v}_t,\boldsymbol{\delta}} (||A\boldsymbol{v}_t\boldsymbol{\delta} + B\boldsymbol{v}_r - \boldsymbol{u}||^2),$$

with the linear transformations $A\boldsymbol{v}_t$ and B in matrix notation, but they allow for confidence weights for each individual flow vector. For a least-squares estimate $\hat{\boldsymbol{v}}_r$ of $\boldsymbol{v}_r$ and a re-formulation that allows to drop $\boldsymbol{\delta}$, this leads to

$$\min_{\boldsymbol{v}_t}(||\boldsymbol{v}_r \circ A^{\perp}(t)^T(B\hat{\boldsymbol{v}}_r(\boldsymbol{v}_t) - \boldsymbol{u})||_2^2).$$

In an expected residual likelihood approach, they directly estimate these confidence weights, based on an assumed Laplacian distribution of the residuals.

[251] first derive a model for a monocular visual-inertial system and aim at making robust state estimations, where the states consist of positions and depths. To this end, the residuals for the visual measurement are minimized, but they are robustified in advance by the function $\rho(r) = I(r \geq 1) + (2\sqrt{r} - 1)I(r < 1)$. On a real-world data set, they achieve real-time performance. See [272] for a similar robustness approach. [372] include 3D to 2D reprojection errors, which enter via the Huber norm.

The Huber loss is also used in [51] where the reprojection error of the estimated trajectory from a linear projection of feature points w.r.t. the estimated trajectory from tracking key points.

[33], [34] use robust loss functions for optical flow estimation. Let $(\boldsymbol{X}_t, y_t)$ be an image point at time t and let $\boldsymbol{v}_t \in \mathbb{R}^2$ be the vector containing the horizontal and vertical image velocity. For the image intensity $I = I(x, y, t)$ of pixel (x, y) at time t, the objective suggested by [33] is

$$\sum_{\boldsymbol{s}=(x,y)} \sum_{\mathcal{R}_s} \rho_1(\partial_x I\boldsymbol{v}_1 + \partial_y I\boldsymbol{v}_2 + \partial_t I, \sigma_1) + \lambda \sum_{i \in \mathcal{N}_s} [\rho_2(\boldsymbol{v}_1^{(\boldsymbol{s})} - \boldsymbol{v}_2^{(\boldsymbol{i})}, \sigma_2) + \rho_2(\boldsymbol{v}_2^{(\boldsymbol{s})} - \boldsymbol{v}_2^{(\boldsymbol{i})}, \sigma_2)]$$

where $\mathcal{N}_s$ containts all neighboring pixels of pixel $\boldsymbol{s}$, where $\mathcal{R}_s$ is some local neighborhood of $\boldsymbol{s}$, for $\sigma_1, \sigma_2, \lambda > 0$, and where ρ_1, ρ_2 are loss functions. The objective has to be optimized w.r.t. $\boldsymbol{v}_1$, $\boldsymbol{v}_1^{(\boldsymbol{s})}$ and $\boldsymbol{v}_2^{(\boldsymbol{s})}$. The first summand encourages the data conservation constraint that the intensity structure of small regions should persist over time, while the second summand encourages the local optical flow of a pixel to be close to that of neighboring pixels. Alternatively, they consider a line-process approach where discontinuities between pixels are modelled separately by binary variables, which leads to a similar objective, and also consider a robust alternative where the truncated squared loss is taken as loss function.

2) LiDAR odometry / point cloud registration

[93] robustify ICP by using the Huber loss function, aiming to minimize the distance

$$\sum_i w_i \min_j(\rho_H(||\tilde{\boldsymbol{p}}_j^* - T(\boldsymbol{\xi})(\tilde{\boldsymbol{p}}_i)||),$$

where the w_i are just indicator variables that take the value one if there is a match between reference and data points.

Welsch's loss function has been applied as a robust error metric for ICP in point cloud registration in [73] in order to quantify the distance between a set of intersection points on the source surface and the target surface, respectively, i.e.,

$$D(\boldsymbol{x}, \boldsymbol{y}) = \rho_W(||\boldsymbol{x} - \boldsymbol{y}||_2^2).$$

[31] consider the objective

$$\min_{R,\vec{\boldsymbol{t}}} \left(\sum_i \rho(||R\tilde{\boldsymbol{p}}_i + \vec{\boldsymbol{t}} - \tilde{\boldsymbol{P}}^*||) \right)$$

for a rotation matrix $R \in \mathbb{R}^{3\times3}$ and a translation vector $\vec{\boldsymbol{t}} \in \mathbb{R}^3$ for point cloud registration and use IRWLS with the Huber, the Tukey or the Cauchy function as robust loss functions ρ.

The Huber loss is applied in [333] for motion-prediction from point clouds, where it is used as motion-prediction loss, spatial and temporal consistency loss.

[125] propose a LiDAR-based direct odometry method with the goal to efficiently find the matching points for the point clouds extracted from the LiDAR data. Direct odometry methods usually compare 2D images, therefore, they first project the 3D LiDAR point to a 2D sphere. As a re-projection of the entire projected 2D image would be time-consuming so that this re-projection is only done on selected key points. Let $\boldsymbol{p}^* \in \mathbb{R}^2$ be the 2D image coordinates of the reference data and let f be a parametric conversion function between sensor and reference data so that $f = f(T(\boldsymbol{\xi}), \boldsymbol{F})$ for a parameter $\boldsymbol{\xi}$ that encodes rotation and translation, a frame $\boldsymbol{F}$, and a translation $T(\boldsymbol{\xi})$ from the Lie group SE(3). Let the residuals from the 2D image coordinate map be $r(\boldsymbol{F}_s, \boldsymbol{F}^*, T(\boldsymbol{\xi}))$ for a sensor frame $\boldsymbol{F}_s$ and a reference frame $\boldsymbol{F}^*$. Then, let a new frame $\boldsymbol{F}_0$ be given with the goal to adjust the corresponding $T_0(\boldsymbol{\xi}_0)$, which is done by minimizing

$$\sum_j \sum_F \rho_H(r(\boldsymbol{F}_s, \boldsymbol{F}^*, T_0(\boldsymbol{\xi}_0)T_j^{-1})$$

at the key points, where the $T_j, j = 1, ..., n$, are frame-specific transformations. Afterwards, the T_j are updated similarly, but where Tukey's biweight loss is used. Experiments on KITTI data and real-world data with an autonomous vehicle confirm real-time performance.

[144] use the Huber loss function when computing functional map matrices that parameterize pairwise correspondences of point clouds in order to better deal with occlusions or deformations. The objectives

$$\sum_{i=1}^{I_{kl}} \rho_H(||\Phi_{l,i}^{(kl)} - \Phi_{k,i}^{(kl)}C||)$$

have to be minimized w.r.t. the map C for all (k, l), which represent the edges in the point cloud graph, and where $\Phi_k^{(kl)}$ are the matrices that represent the matched points from point cloud $\mathcal{P}_k$ to point cloud $\mathcal{P}_l$, for the number I_{kl} of matches.

The squared loss in the ICP algorithm has been replaced with the LMS criterion in [375] and [214], and with the LTS criterion for example in [59] and [243]. [245] proposed the so-called fractional root mean squared distance as distance measure for ICP, which is essentially an LTS criterion, up to taking the square root. [145] consider a truncated absolute loss. [115] propose a differentiable variant of the Huber loss. [68] use the family of parametrized robust loss functions

from [23] and propose an algorithm where one alternatingly optimizes for the parameters of this loss function and the actual regression parameter. [346] propose a graduated non-convexity approach where a non-convex robust loss is optimized by iteratively optimizing a sequence of surrogates, which are initially convex but gradually become non-convex. This method is applied to point cloud registration with the Geman-McClure and the truncated least squares criterion.

[193] consider matching a data and a reference point cloud, resulting in the objective

$$\text{argmin}_{R\in\text{SE}(3),\vec{t}\in\mathbb{R}^3}\left(\sum_i\sum_j \rho(r(\tilde{\boldsymbol{p}}_j^* - R\tilde{\boldsymbol{p}}_i - \vec{t}))\right)$$

for a robust loss function ρ. Experiments on different data sets confirm a total computational time of less than one second, confirming real-time performance.

A similar approach has been used in [335] for point-to-plane matching, where the distance between a point from the point cloud and the nearest point from a local plane on the map. They use the Huber loss function. In [376], the truncated least squares loss is used in order to find a transformation that aligns points from a LiDAR frame with points from a local map for ego-motion estimation. Experiments on KITTI data and real-world data collected from a robot confirm real-time performance of their overall LiDAR-only odometry and mapping pipeline.

[143] propose a loss function that can be interpreted as a soft counterpart of a truncated least squares loss, namely $\rho(\boldsymbol{r},k,w) = w^2||\boldsymbol{r}||^2 + (1-w)^2k^2$. In other words, each residual is accompanied with a weight which decides the trade-off between the squared loss and a constant loss. This loss function is not used directly as objective for the estimation of the optimal transformation $T \in \text{SE}(3)$ but as a penalty term, i.e., the objective is

$$\sum_i ||(\boldsymbol{r})^{(i)})^*||^2 + \sum_j \rho(w_j, ||\boldsymbol{r}^{(j)}||^2),$$

where $(\boldsymbol{r}^{(i)})^*$ denotes the residuals w.r.t. a reference point cloud and where $\boldsymbol{r}^{(j)}$ denotes residuals from the LiDAR point cloud. The objective is optimized w.r.t. the weights and the transformation alternatingly. The optimization w.r.t. the weights leads to the closed-form solution $w_j^* = k^2(||\boldsymbol{r}^{(j)}||^2 + k^2)^{-1}$, implying the loss $\rho(||\boldsymbol{r}^{(j)}||^2,k,w_j) = k^2||\boldsymbol{r}^{(j)}||^2(k^2 + ||\boldsymbol{r}^{(j)}||^2)$. This is just a scaled Geman-McClure loss. They achieve real-time performance on different real data sets. A similar objective function has been proposed in [373]. On real-world data from urban areas in Hong Kong, they achieve real-time performance.

[330] use the Huber kernel loss as loss function in laser localization. The objective is then

$$\sum_i w_i\rho_H(r(T(\boldsymbol{\xi})(\tilde{\boldsymbol{p}}_i),\tilde{\boldsymbol{P}}^*)),$$

where w_i an indicator which is zero if $\tilde{p}_i$ is considered to be an outlier, which is done by comparing the median of the error of the posterior predictive corresponding to this point with the population median of the error.

3) Navigation/tracking via filtering

An important class of state estimation techniques are **Kalman filters** (KF). The linear KF assumes a state space model of the form (e.g., [269])

$$\boldsymbol{X}_t = F_t\boldsymbol{X}_{t-1} + \boldsymbol{\nu}_t, \quad \boldsymbol{Y}_t = Z_t\boldsymbol{X}_t + \boldsymbol{\epsilon}_t,$$

with transition matrices $F_t \in \mathbb{R}^{p\times p}$, $Z_t \in \mathbb{R}^{q\times p}$, and noise variables $\boldsymbol{\epsilon}_t \sim \mathcal{N}_q(\boldsymbol{0}_q, V_t)$, $\boldsymbol{\nu}_t \sim \mathcal{N}_p(\boldsymbol{0}_p, Q_t)$. The first equation is the state equation, describing the evolution of the states of the system, while the second equation is the measurement equation that describes the generation of noisy measurement outputs from the underlying true states. In control theory, one would also include a controller input in the state equation (see Sec. V-C). The state space model described here is time-discrete and time-variant. In the less general time-invariant settings, one has static transition matrices F and Z. The goal in Kalman filtering is to estimate the true states $\boldsymbol{X}_t$ when measuring the $\boldsymbol{Y}_t$. There are several ways how to robustify the KF, for example, by robustifying the loss function is the least-squares interpretation of the KF, by assuming a different noise distribution that is capable to model large errors which would appear as outliers under the Gaussian assumption, or outlier detection. In this paper, since we are not aware of any robust approach for autonomous driving in a continuous-time setting, we always have a discrete-time setting.

[269] distinguish between **additive outliers (AOs)**, which affect the observations, i.e.,

$$\boldsymbol{\epsilon}_t^{\text{re}} \sim (1-r_{AO})\mathcal{L}(\boldsymbol{\epsilon}_t^{\text{id}}) + r_{AO}\mathcal{L}(\boldsymbol{\epsilon}_t^{ru}),$$

and **innovation outliers (IO)**, which affect the innovations, i.e.,

$$\boldsymbol{\nu}_t^{\text{re}} \sim (1-r_{IO})\mathcal{L}(\boldsymbol{\nu}_t^{\text{id}}) + r_{IO}\mathcal{L}(\boldsymbol{\nu}_t^{ru}),$$

where $\mathcal{L}(\boldsymbol{\epsilon}_t^{\text{id}})$, $\mathcal{L}(\boldsymbol{\epsilon}_t^{ru})$, $\mathcal{L}(\boldsymbol{\nu}_t^{\text{id}})$, $\mathcal{L}(\boldsymbol{\nu}_t^{ru})$ denote the distributions of the ideal and contaminated noise terms, respectively, and where r_{AO}, $r_{IO} \in [0,1]$ are the respective contamination radii. The main difference is that IOs affect subsequent states, covering level shifts or linear trends, so that not only the current observation is affected. One can also define **substitutive outliers (SO)**, which directly manipulate the observation distribution and not the noise distribution, i.e.,

$$\boldsymbol{Y}_t^{\text{re}} \sim (1-r_{SO})\mathcal{L}(\boldsymbol{Y}_t^{\text{id}}) + r_{SO}\mathcal{L}(\boldsymbol{Y}_t^{ru}).$$

The Kalman filtering algorithm, going back to [164], is given by the following recursive scheme (here, in the notation of [269]): Initialization

$$\boldsymbol{X}_{0|0} = \boldsymbol{a}_0, \quad \Sigma_{0|0} = Q_0, \tag{4}$$

prediction

$$\boldsymbol{X}_{t|t-1} = F_t\boldsymbol{X}_{t-1|t-1}, \quad \Sigma_{t|t-1} = F_t\Sigma_{t-1|t-1}F_t^T + Q_t, \tag{5}$$

and correction

$$\boldsymbol{X}_{t|t} = \boldsymbol{X}_{t|t-1} + K_t\boldsymbol{\Delta}\boldsymbol{Y}_t, \quad \Sigma_{t|t} = (I_p - K_tZ_t)\Sigma_{t|t-1}, \tag{6}$$

for

$$\Delta X_t = X_t - X_{t|t-1}, \quad \Delta Y_t = Y_t - Z_t X_{t|t-1} = Z_t \Delta X_t + \epsilon_t, \tag{7}$$

and

$$\Delta_t = Z_t \Sigma_{t|t-1} Z_t^T + V_t, \quad K_t = \Sigma_{t|t-1} Z_t^T \Delta_t^-. \tag{8}$$

Here, the quantity K_t is referred to as the Kalman gain.

This recursive scheme can also be interpreted as a least-squares approach (e.g., [288]). In the notation of [65], denoting

$$\begin{pmatrix} Y_t \\ X_{t|t-1} \end{pmatrix} = \begin{pmatrix} Z_t \\ I \end{pmatrix} X_t + \begin{pmatrix} \epsilon_t \\ r_t \end{pmatrix}$$

for

$$r_t = X_{t|t-1} - X_t,$$

one can compactly write

$$\tilde{Y}_t = \tilde{Z}_t X_t + \tilde{r}_t,$$

where $\tilde{r}_t$ has a block diagonal covariance matrix $\tilde{R}_t$. The estimation of the states via a squared loss leads to a least-squares solution with prediction

$$\hat{X}_{t|t} = (\tilde{Z}_t^T \tilde{R}_t^{-1} \tilde{Z}_t)^{-1} \tilde{Z}_t^T \tilde{R}_t^{-1} \tilde{Y}_t. \tag{9}$$

In non-linear dynamics, suitable versions of the linear KF have been proposed in the literature, where the state-space model is given by

$$X_t = f(X_{t-1}) + \nu_t, \quad Y_t = h(X_t) + \epsilon_t,$$

for differentiable functions f and h. In the recursive KF scheme however, the quantities F_t and Z_t are required. For the EKF, a first-order linearization of f is done at $X_{t-1|t-1}$, while a linearization of h is done at $X_{t|t-1}$.

The **unscented Kalman filter** (UKF) also allows for non-linear transformations but does not perform a linear approximation as the EKF. Instead, a so-called unscented transformation [162], [312] in order to approximate the posterior mean and variance of the underlying function is computed.

The **cubature Kalman filter** (CKF) uses the radial-spherical cubature rule [14] instead of the unscented transformation as in the UKF in order to estimate the posterior mean and variance.

The following approaches consider robust loss functions.

[49] propose to replace the squared loss of the linear KF by the maximum correntropy (MMC) criterion, which is a local similarity measure and therefore insensitive to large outliers. The MMC criterion is given by

$$\int \kappa(x,y) dF(x,y)$$

for a shift-invariant Mercer kernel κ, e.g., the Gaussian kernel $\kappa(x,y) = G_\sigma(r) = \exp(-r^2/(2\sigma^2))$. Given residuals r_t, one can therefore estimate the correntropy by the arithmetic mean of the $G_\sigma(r_t)$. In the KF context, they define the errors

$$\tilde{\epsilon}_t = \begin{pmatrix} -(X_t - \hat{X}_{t|t-1}) \\ \epsilon_t \end{pmatrix}$$

and denote $\mathbb{E}[\tilde{\epsilon}_t \tilde{\epsilon}_t^T] = B_t B_t^T$ with a matrix B_t that can be computed by a Cholesky decomposition of $\mathbb{E}[\tilde{\epsilon}_t \tilde{\epsilon}_t^T]$. It follows that $D_t = W_t X_t + r_t$ for

$$D_t = B_t^{-1} \begin{pmatrix} \hat{X}_{t|t-1} & Y_t \end{pmatrix}, \quad W_t = B_t^{-1} \begin{pmatrix} I \\ H_t \end{pmatrix}, \quad r_t = B_t^{-1} \tilde{\epsilon}_t, \tag{10}$$

where r_t is white noise. Now, the correntropy objective leads to

$$\hat{X}_{t|t-1} = \text{argmax}_x \left(\frac{1}{p+q} \sum_{i=1}^{p+q} G_\sigma((D_t)_i - (W_t)_i x) \right).$$

They derive an iterative fixed-point algorithm in order to find the optimal solution and prove a sufficient condition for convergence.

A CKF based on the MMC criterion has been applied for cooperative localization of underwater vehicles. [378] apply an MCC-based cubature information filter for tracking aerial unmanned vehicles. In their numerical simulations with a step size of 1s, their filter requires a computation time of around 0.2s. [293] propose to combine a UKF with the MCC criterion based on the t-kernel with an additional weighting scheme in order to safeguard the estimation against extreme outliers. As weight function, applied to the individual components of the states, they consider the biweight, Huber, Hampel and Andrews function. On real-world data collected from an autonomous underwater vehicle, their algorithm achieves real-time performance. [195] use a KF with the MCC criterion for pseudorange estimation. They consider localizing and tracking in real-world experiments and achieve similar computational efficiency than the standard KF.

[185] consider collaborative localization and propose an EKF updating scheme with the MCC. Here, as for the required Mercer kernel, they consider a Cauchy kernel. [88] consider a Laplacian kernel and apply the resulting MCC-based EKF for cooperative localization of autonomous underwater vehicles. In their simulations, the computation time was around twice as much as for the standard EKF.

An alternative loss function for robust state estimation with the KF has been proposed by [72] who consider the residual least entropy-like loss function

$$\begin{aligned} H_k(D_k, q_1, ..., q_k) \\ = I(D_k \neq 0) \cdot \frac{-1}{\ln(k)} \sum_{i=1}^{k} q_i \ln(q_i), \\ D_k = \sum_{i=1}^{k} ||r_i||^2, \\ q_i = \frac{||r_i||^2}{\sum_{j=1}^{k} ||r_j||^2}, \end{aligned}$$

for the residuals r_i. This loss function is used as a penalty term for the weighted least-squares objective which encourages a large entropy of the residuals and hence many small and a few large residuals.

[65] consider GNSS/INS integration (global navigation satellite system/ intertial navigation system) and propose a robust KF by robustifying the update step with an M-estimator. They point out that robust methods require a larger number of measurements than classical ones (which is a consequence of the efficiency loss, since the least-squares estimator is the maximum likelihood estimator under Gaussian noise and hence achieves maximum efficiency) but that GNSS applications usually do not have many satellites in view. They consider the standardized version W_t of $\tilde{R}_t^{-1}$ from Eq. 9 by standardizing the measurements first. Then, W_t is updated using IRWLS, which is robustified by the Huber estimation

$$(W_t)_{ii} = w_H([\bar{\boldsymbol{Y}}_t - \bar{Z}_t\hat{\boldsymbol{X}}_{t|t}]_{ii})$$

for the respective standardized measurements $\bar{\boldsymbol{Y}}_t$ and $\bar{Z}_t$. In their experiments, they consider different types of contamination, which are single biases, multiple biases, and ramps.

[97] consider, in addition to innovation and observation outliers, so-called **structural outliers**, i.e., where the linear mapping Z_t resp. F_t in the state space model may be misspecified. They assume that the observations are synchronized, otherwise, delayed observations may be treated as outlying data. First, they aim at robustly estimating the covariance matrix in the least-squares interpretation of the KF. To this end, they consider the Stahel-Donoho estimator

$$\sup_{||\boldsymbol{u}||=1}\left(\frac{|\boldsymbol{h}_t^T\boldsymbol{u} - \text{med}_j(\boldsymbol{h}_j^T\boldsymbol{u})|}{\text{MAD}_j(\boldsymbol{h}_j^T\boldsymbol{u})}\right)$$

for the data points h, which are here the matrices $(Z_t^T, I)^T$. However, they argue that applying the estimator in each time step would detect only structural outliers and therefore propose to use the vector $\tilde{\boldsymbol{Y}}_t = (\boldsymbol{Y}_t^T, \hat{\boldsymbol{X}}_{t|t-1}^T)^T$ instead of the $\boldsymbol{h}_t$ as it already captures the effects of all three types of outliers. Points whose value s of the Stahel-Donoho estimator is larger than some threshold τ are downweighted in the sense that their new weight is τ^2/s^2, where $\tau = 1.5$ in [97]. As structural outliers appear as leverage points in the least-squares interpretation of the KF, a GM-estimator with the Huber loss function and the weights arising from the Stahel-Donoho estimator is applied, i.e., one minimizes

$$\sum\nolimits_i w_i\rho_H(\tilde{r}_i),$$

with the residuals $\tilde{r}_i = r_i s^{-1} w_i^{-1}$ with the MAD s of the vector of residuals r_i. Due to non-linearity, this problem is solved using IRWLS. Finally, the update filter error has to be adapted. They therefore compute the IC of the GM-estimator, given by

$$\text{IC}(x, \boldsymbol{X}, P) = \frac{\psi(\tilde{x})}{\mathbb{E}_P[\psi'(x)]}(A^TA)^{-1}\boldsymbol{X}w$$

for $\tilde{x} = xs^{-1}w^{-1}$, which enables to compute the asymptotic covariance matrix $\Sigma_{t|t} = \mathbb{E}_P[\text{IC}(x, \boldsymbol{X}, P)(\text{IC}(x, \boldsymbol{X}, P))^T]$.

[364] propose distributionally robust filtering, where a minimax problem is solved. Let $\boldsymbol{Z}_t = (\boldsymbol{X}_t, \boldsymbol{Y}_t)$ and let $F_{\boldsymbol{Z}_t|\boldsymbol{Y}_{t-1}}$ denote the conditional joint state-measurement distribution at time step t. The new state is estimated by

$$\hat{X}_{t+1} = \min_{\boldsymbol{s}}(\max_{F\in\mathcal{U}(F_{\boldsymbol{Z}_t|\boldsymbol{Y}_{t-1}})}(\mathbb{E}[(\boldsymbol{X}_t - \boldsymbol{s})(\boldsymbol{X}_t - \boldsymbol{s})^T])),$$

for an uncertainty set $\mathcal{U}(F_{\boldsymbol{Z}_t|\boldsymbol{Y}_{t-1}})$ around $F_{\boldsymbol{Z}_t|\boldsymbol{Y}_{t-1}}$. In their algorithm, they construct this set by mean and covariance constraints around a nominal distribution. The problem can be re-written as a nonlinear semi-definite program. In their experiments, they consider tracking a hypersonic vehicle.

[284] replace the least-squares regression problem arising in the KF by a minimax problem. In particular, they propose to minimize the worst-case expected squared residuals over an uncertainty set. As for this uncertainty set, they either use all normal distributions with the same mean as the ideal distribution but whose covariance lies within a certain radius around the ideal covariance, or a Wasserstein-based contamination ball containing all normal distributions whose W_2-distance from the ideal distribution is bounded by the contamination radius. On real-world data, the performance of their algorithm is comparable to that of the standard EKF, thus allowing for real-time performance.

[74] consider aircraft ground inspection, which is vulnerable to large positioning errors of GNSS. They consider a robust EKF based on M-estimation. They essentially robustify the KF in the same manner as [49], but where the $\boldsymbol{r}_t$ from Eq. 10 do not enter the correntropy criterion but a weighted least squares objective, i.e.,

$$\min_{\boldsymbol{x}}(\boldsymbol{r}_t^T(\boldsymbol{x})W\boldsymbol{r}_t(\boldsymbol{x})),$$

for the weight matrix $W = \text{diag}(w_H(\boldsymbol{r}_{t,i}))$ with the Huber weight function w_H. [75] propose a grid search in order to select the hyperparameter K of the Huber function in a data-driven way, according to the horizontal accuracy. More precisely, the hyperparameter is chosen according to a difficulty level of the scenario, and this level is predicted using a NN. In order to make the predictions more interpretable, [76] replace the NN by a SVM.

[28] propose a robustification of the EKF by downweighting measurement outliers. They use the formulation

$$\hat{\boldsymbol{X}}_t = \text{argmin}_{\boldsymbol{X}_t}(||\boldsymbol{X}_t - f(\hat{\boldsymbol{X}}_{t-1})||^2_{\hat{Y}_t} + ||\boldsymbol{Y}_t - h(\boldsymbol{X}_t)||^2_{\tilde{Q}_t})$$

for $\tilde{Q}_t = \hat{Q}_t^{1/2}W^{-1}\hat{Q}_t^{T/2}$. Here, $Q_t^{1/2}$ is the Cholesky factorization of Q_t. The weight matrix W is given by $W = \text{diag}(w(Q_t^{1/2}(\boldsymbol{Y}_t - h(\boldsymbol{X}_t))))$ for a weight function w corresponding to a robust loss function. In their experiments, they use the Huber weight function. [270] integrate feature maps into the EKF for GNSS positioning. Those feature maps contain information about, for example, satellite visibility or spatio-temporal features, allowing for a prior distribution of the pseudorange residuals. In the EKF, observations whose pseudorange residual deviates considerably from the expected ones are downweighted. This is done by applying the weight function corresponding to a robust loss function, for which the Huber, Tukey and Geman-McClure loss function are considered, to the predicted pseudorange residuals.

[119] apply parallel robust EKFs for a Bayesian approach for robust localization from GNSS data. Instead of the Gaussian likelihood, they consider densities of the form $p(\boldsymbol{Y}_t|\boldsymbol{X}_t) \propto \exp(-\rho(\boldsymbol{r}_t))$ for the residual $\boldsymbol{r}_t = \boldsymbol{Y}_t - \boldsymbol{h}(\boldsymbol{X}_t)$ and where ρ is the Huber or the Tukey loss function. The covariance matrix V_t of the measurement equation is updated via $\tilde{R}_t = (\psi(\boldsymbol{r}_t))^{-1}R_t$ for $\psi(\boldsymbol{r}_t) = \partial_{\boldsymbol{r}}\rho(\boldsymbol{r})|_{\boldsymbol{r}=\boldsymbol{r}_t}$. However, due to multi-modal uncertainties in the measurements, single Gaussian distributions are not suitable. To this end, they replace the standard posterior,

$$p(\boldsymbol{X}_t|\boldsymbol{Y}_{1:t}) \propto p(\boldsymbol{Y}_t|\boldsymbol{X}_t)\int p(\boldsymbol{X}_t|\boldsymbol{X}_{t-1})p(\boldsymbol{X}_{t-1}|\boldsymbol{Y}_{1:t-1})d\boldsymbol{X}_{t-1},$$

with

$$p(\boldsymbol{X}_t|\boldsymbol{X}_t^l, \boldsymbol{Y}_{1:t}) \propto p(\boldsymbol{Y}_t|\boldsymbol{X}_t^l, \boldsymbol{X}_t)\int p(\boldsymbol{X}_t|\boldsymbol{X}_t^l, \boldsymbol{X}_{t-1})p(\boldsymbol{X}_{t-1}|\boldsymbol{X}_t^l, \boldsymbol{Y}_{1:t-1})d\boldsymbol{X}_{t-1},$$

for linearization points $\boldsymbol{X}_t^l$ at which the EKF transition and the robust loss ρ are linearized. As the selection of linearization points is accompanied by uncertainties itself, [119] propose the update rule

$$p(\boldsymbol{X}_t^l, \boldsymbol{Y}_{1:t}) \propto p(\boldsymbol{Y}_t|\boldsymbol{X}_t^l)\int p(\boldsymbol{X}_t^l|\boldsymbol{X}_{t-1}^l)p(\boldsymbol{X}_{t-1}^l|\boldsymbol{Y}_{1:t-1})d\boldsymbol{X}_{t-1}^l.$$

In their algorithm, one starts with an initial set of linearization points and iteratively updates this whole set and corresponding weights that are computed via the posterior $p(\boldsymbol{Y}_t|\boldsymbol{X}_t^l)$, so that the distribution $p(\boldsymbol{X}_t|\boldsymbol{Y}_{1:t})$ can finally be estimated using Rao-Blackwellization.

The UKF has been robustified by Huberization in [337]. Given the non-linear dynamics

$$X_{t+1} = f(\boldsymbol{X}_t) + \boldsymbol{\nu}_t, \quad \boldsymbol{Y}_t = h(\boldsymbol{X}_t) + \boldsymbol{\epsilon}_t,$$

one can write

$$\begin{pmatrix}\boldsymbol{Y}_{t+1}\\ \boldsymbol{X}_{t+1|t}\end{pmatrix} = \begin{pmatrix}h(\boldsymbol{X}_{t+1})\\ \boldsymbol{X}_{t+1}\end{pmatrix} + \begin{pmatrix}\boldsymbol{\epsilon}_t\\ \boldsymbol{\Delta}\hat{\boldsymbol{X}}_{t+1|t}\end{pmatrix}$$

for the predicted state $\hat{\boldsymbol{X}}_{t+1|t}$ at time $(t+1)$ and its error $\boldsymbol{\Delta}\hat{\boldsymbol{X}}_{t+1|t}$. For the covariance V_t of $\boldsymbol{\epsilon}_t$ and the covariance $\Sigma_{t+1|t}$ of $\hat{\boldsymbol{X}}_{t+1|t}$, one computes

$$\tilde{\boldsymbol{Y}}_{t+1} = S_{t+1}^{-1/2}\begin{pmatrix}\boldsymbol{Y}_{t+1}\\ \hat{\boldsymbol{X}}_{t+1|t}\end{pmatrix}, \quad S_{t+1} = \begin{pmatrix}V_t & 0\\ 0 & \Sigma_{t+1|t}\end{pmatrix},$$

$$g(\boldsymbol{X}_{t+1}) = S_{t+1}^{-1/2}\begin{pmatrix}h(\boldsymbol{X}_{t+1})\\ \boldsymbol{X}_{t+1}\end{pmatrix}, \quad \boldsymbol{\xi}_{t+1} = S_{t+1}^{-1/2}\begin{pmatrix}\boldsymbol{\epsilon}_t\\ \boldsymbol{\Delta}\hat{\boldsymbol{X}}_{t+1|t}\end{pmatrix},$$

so that $\tilde{\boldsymbol{Y}}_{t+1} = g(\boldsymbol{X}_{t+1}) + \boldsymbol{\xi}_{t+1}$ holds. The objective for finding the prediction $\hat{\boldsymbol{X}}_{t+1|t}$ is then

$$\min_{\boldsymbol{x}}\left(\sum_{i=1}^{p+q}\rho_H(r_{t+1,i}(\boldsymbol{x}))\right),$$

where the residuals are given by $\boldsymbol{r}_{t+1}(\boldsymbol{X}_{t+1}) = \tilde{\boldsymbol{Y}}_{t+1} - g(\boldsymbol{X}_{t+1})$. [337] use this Huberized UKF for underwater terrain matching, where $\boldsymbol{X}_t$ represents the 2D coordinates of the vehicle. [46] apply it to tracking.

[322] propose an adaptive variant of the robust UKF with an application in vehicle tracking. They consider the dynamics model error and the measurement model error simultaneously by treating the respective residuals separately, i.e.,

$$\min_x(\lambda_t||\boldsymbol{X}_t - \hat{\boldsymbol{X}}_{t|t-1}||^2_{\Sigma_{t|t-1}} + ||\boldsymbol{F}_t\boldsymbol{X}_t - \boldsymbol{Y}_t||^2_{V_t^{-1}})$$

with a fading factor λ_t which is computed by

$$\lambda_t = \begin{cases}1, & |\boldsymbol{\Delta}\tilde{\boldsymbol{X}}_t| \le \tau\\ \frac{\tau}{|\boldsymbol{\Delta}\tilde{\boldsymbol{X}}_t|}, & |\boldsymbol{\Delta}\tilde{\boldsymbol{X}}_t| > \tau\end{cases}, \quad \boldsymbol{\Delta}\tilde{\boldsymbol{X}}_t = \frac{||\tilde{\boldsymbol{X}}_t - \hat{\boldsymbol{X}}_{t|t-1}||}{\sqrt{\mathrm{tr}(\Sigma_{t|t-1}^{-1})}},$$

for some threshold $\tau > 0$.

[21] alternatingly use the Huber and the loss corresponding to the dynamically scaled covariance approach from [3] in the EKF and consider the navigation of unmanned underwater vehicles. On real-world data, the computation time is close to that of the standard KF, with at most around 25% overhead. [22] modify the EKF by alternatingly optimizing the MSE and the Huber loss in the sense that in a first iteration, $\hat{\boldsymbol{X}}_{t|t}$ is estimated using the updating steps corresponding to the Huber loss. Then, $\hat{\boldsymbol{X}}_{t|t}$ enters as prior $\hat{\boldsymbol{X}}_{t|t-1}$ in the updating steps corresponding to the MSE. The average runtime on real-world data is around 75% higher than for the standard KF.

See further applications of Huberized Kalman Filters for spacecraft attitude estimation (linear KF; [166]), elliptical orbit rendezvous and docking (EKF; [167]), navigation (UKF; [47], [250], CKF; [307]), vehicle tracking (CKF; [124], [197]), underwater tracking (EKF; [82]), and collaborative localization (EKF; [129]).

In contrast to loss-based filters where a robustification of the loss function is done, **noise modeling and covariance scaling** approaches consider heavy-tailed distributions, assuming that the measurements can be contaminated by heavy-tailed noise, in contrast to the standard KF that assumes Gaussian noise. This idea essentially goes back to [328]. It has been shown in [267] that in the ideal, i.e., Gaussian, model, the usage of a t-distribution with small degrees of freedom leads to a high efficiency loss. In particular, using variational approximation allows for learning the real noise distribution in an online manner, even allowing for non-stationary (e.g., due to changing environments as argued in [152]) loss distributions.

[2] propose a structured variational approach where they assume an inverse Wishart distribution of the covariance matrix V_t. As they assume that $\boldsymbol{Y}_t|\boldsymbol{X}_t, R_t \sim \mathcal{N}(F\boldsymbol{X}_t, R_t)$, marginalizing out R_t leads to a t-distribution as the conditional distribution of $\boldsymbol{Y}_t|\boldsymbol{X}_t$. They derive that the marginal log-likelihood of the $\boldsymbol{Y}_t$ can be expressed as the sum of a lower bound of the marginal likelihood of the data and the KL-

divergence between the true and the approximate posterior distribution of $(\boldsymbol{X}_t, R_t)$ given $\boldsymbol{Y}_t$. For iid. noise and for a slowly-drifting noise model, where the two parameters of the inverse Wishart distribution of R_t themselves obey a first-order model, they derive an algorithm in order to compute an approximate posterior. They apply their method for GPS position estimation of a car. [1] assume

$$\boldsymbol{X}_t|\boldsymbol{X}_{t-1} \sim \mathcal{N}(F^T\boldsymbol{X}_{t-1} + \boldsymbol{b}, Q), \quad \boldsymbol{Y}_t|\boldsymbol{X}_t, V_t \sim \mathcal{N}(Z^T\boldsymbol{X}_t + \boldsymbol{d}, V_t)$$

where the observations noise V_t^{-1} is assumed to follow a Wishart distribution, again leading to a t-distribution of $\boldsymbol{Y}_t|\boldsymbol{X}_t$. The posterior $p(\boldsymbol{X}_t|\boldsymbol{Y}_1, .., \boldsymbol{Y}_t)$ is approximated by structured variational filtering. They apply their method for position estimation from GPS data.

The approach from [1] has been extended to the nonlinear case in [246], [273]. [150] argue that these such variational Bayes approaches as in [273] can handle slowly time-varying measurement noise covariance matrices V_t, but that they assume accurate estimation of the process noise covariance matrices Q_t, otherwise, their performance decreases. Therefore, they propose to assume inverse Wishart priors for both V_t and the prediction error covariance matrix P_t, which, by the prediction step $\boldsymbol{X}_{t|t-1} = F_{t-1}\boldsymbol{X}_{t-1|t-1}$, satisfies

$$P_{t|t-1} = F_{t-1}P_{t-1|t-1}F_{t-1}^T + Q_{t-1}.$$

The new states are then inferred jointly with $P_{t|t-1}$ and V_t via variational approximation. Their method is applied to target tracking.

[233] assume that the measurement noise is skewed t-distributed and approximate the posterior observation distribution with variational inference. They apply their filter for GNNS position estimation. In their experiments, they report that the computational time of their filter exceeds that of the standard KF by a factor of around 5 to 10. [374] consider GNSS positioning and assume t-distributed measurement noise. They propose to estimate the degrees of freedom outside the variational Bayes iteration via inversely scaling a baseline degree of freedom with the Mahalanobis distance of the current innovation to a Gaussian distribution with the current innovation covariance matrix. The method is applied in a field test for position estimation of a vehicle in Beijing. [55] assume t-distributed process noise, while the measurement noise is assumed to be Gaussian. They consider SINS navigation in a real-world car-mounted experiment. [158] model the measurement noise by a convex contamination model, where the ideal Gaussian distribution is contaminated with a Gaussian distribution with a different covariance matrix. They allow the contamination radius to vary in time. As they consider a situation where the states are observed by multiple agents, they have an observation equation for each agent j with individual transition matrices Z_t^j and noises $\boldsymbol{\epsilon}_t^j$. In a sliding window approach, the joint posterior of the states, the process covariance matrices, the agent-specific measurement covariance matrices and contamination radii is approximated via variational inference. They apply their method in a target tracking simulation with multiple sensors. [170] argue that a robust filter is less efficient than a filter with the Gaussian assumption and propose to use two models, one with Gaussian and one with t-distributed measurement noise, and to combine them using Bayesian model averaging. They use their method for target tracking. In their simulations, they observe a computational time of their filter of around twice as high as for the standard CKF, while only requiring around 40% of the time of the CKF where iterative variational Bayes approximations are used. [126] propose to use α-stable sub-Gaussian distributions for the measurement noise in the linear KF, and compare noise modeling with low- and heavy-tailed noise distributions, such as α-stable sub-Gaussians, Gaussian mixtures or t-distributions. [336] a mixture of a Gaussian and a Gaussian inverse-Gamma distribution for the measurement noise and apply their algorithm to the navigation of an underwater vehicle. [191] invoke a Gaussian-exponential distribution for the measurement noise.

[148] propose a Gaussian-inverse-Wishart mixture distribution for the state transition. They argue that such a mixture has the advantage over a single Gaussian-inverse-Wishart distribution when only inaccurate prior information is available. As for the prior, they assume a Dirichlet distribution. The conditional observation distribution, they assume a Gaussian-inverse-Wishart distribution and derive a variational approximation algorithm of the joint posterior of the current and previous state, the measurement and the state covariance matrices and the mixing parameters. They apply their algorithm to target tracking. [86] track an unmanned surface vehicle by assuming an inverse Wishart distribution for the measurement covariance matrix and computing the posterior distribution of the states and covariances by variational inference. [207] first scale the measurement covariance matrix with IGGIII weights, and propose to model the covariance matrix with variational Bayes and an inverse Wishart distribution as prior.

[146] argue that the estimation accuracy of the prediction error covariance matrix, $\Sigma_{t|t-1}$, depends on the state noise covariance matrix, Q_t. Therefore, if only inaccurate prior information about the latter is available, they propose to not use the one-step prediction-error covariance matrix directly in their variational Bayesian adaptive KF algorithm, but estimate a prior scale matrix via the EM algorithm. They apply their strategy for collaborative localization with two surface vehicles and one autonomous underwater vehicle.

[66] use the CKF with a sigma-point update rule for GNSS/INS estimation. In order to deal with measurement outliers, they propose to include switching variables which are Bernoulli-distributed, where the presence of an outlier would correspond to the value 0, with Beta prior. The measurement covariance matrix is scaled with the inverse of the expectation of the switching variable in the respective time step. The joint posterior of the states, switching variables and their priors are updated via variational inference. They apply their method in a real-world experiment with car-mounted GNSS/INS.

[147] assume that both the process and measurement noise are t-distributed. In a smoothing approach with nonlinear dynamics, they infer the trajectory in a fixed time window via variational inference of the joint posterior, and apply their technique to target tracking. [317] consider car tracking and assume t-distributed process noise and a Gaussian-generalized hyperbolic distribution for the measurement noise in the EKF. The latter is a mixture of Gaussian distributions, where the mixture distribution is a generalized inverse Gaussian distribution, thus a joint posterior of the states, covariances, distribution and mixture parameters is computed by variational approximation. [252] consider target tracking and assume a Gaussian-exponential-Gamma distribution for both the process and the measurement noise.

[152] propose a Gaussian-Student's t mixture distribution (GSTM) in order to address non-stationary, heavy-tailed noise distributions for both the states and the observations. The GSTM distribution is of the form

$$p(\boldsymbol{x}|\pi) = \pi\mathcal{N}(\boldsymbol{x}, \boldsymbol{\mu}, \Sigma) + (1-\pi)t(\boldsymbol{x}, \boldsymbol{\mu}, \Sigma, \nu),$$

where π is the mixing parameter, which has to be inferred and for which a Beta distribution is assumed as prior distribution. While the GSTM distribution has lighter tails than the respective pure t-distribution, it has heavier tails than the respective Gaussian distribution. [152] argue, based on the influence function of the GSTM distribution, which is close to that of the Gaussian distribution in a vicinity of the mean and tends to that of the t-distribution outside, that the GSTM distribution has the same efficiency as the Gaussian distribution on clean data and the same efficiency as the t-distribution on contaminated data. The joint posterior distribution of the states, the mixing parameters and the degrees of freedom is approximated via a variational Bayes approach.

[314] assume time-varying skewness in the measurement noise, which they argue to result from imperfect synchronization and a variable nonline of sight. They propose a so-called shape-parameter mixture distribution of the measurement noise, which is a mixture of Gaussian scale mixture distributions w.r.t. the shape parameters, extending the work of [149] who initially proposed the pure Gaussian scale mixture for the process and the measurement noise. As for the mixing prior, they assume a Dirichlet distribution. They apply their algorithm to robot tracking.

[209] incorporate both heavy-tailed measurement noise and inequality constraints in a variational Bayes algorithm. First, they assume a skewed t-distribution for the measurement noise and an inverse Wishart distribution for the predicted error covariance matrix. As for the inequality constraints, they consider linear constraints of the form $\boldsymbol{a}_t \leq D_t\boldsymbol{X}_t \leq \boldsymbol{b}_t$, for some constraint matrix D_t. These constraints are integrated into the variational approximation via truncation of one element of the predicted state, conditioning the computed distribution onto the feasible set. In their experiments, they track a mobile robot.

[356] consider target tracking and extend the state-space model by multiplicative noise in the measurement equation, leading to the model

$$\boldsymbol{X}_t = F_t\boldsymbol{X}_{t-1} + \boldsymbol{\nu}_t, \quad \boldsymbol{Y}_t = m_t Z_t \boldsymbol{X}_t + \boldsymbol{\epsilon}_t.$$

They motivate the multiplicative term m_t by the multipath effect as well as fading and scattering when considering underwater acoustics. The two additive and the multiplicative noise are modelled as generalized t-distributions, and the posteriors are approximated by variational inference.

There is further literature where variational filtering is used in situations where multiple state-space models have to be considered, for example, in sensor fusion, collaborative navigation or centralized estimation settings.

[281] consider state estimation of unmanned surface vehicles and propose to perform the estimation remotely in order to save onboard computational capacities. To this end, they propose a stochastic event-triggered communication strategy. Let $\boldsymbol{Y}^{\text{old}}$ be the most recent observation that the USV transmitted to the remote station. For each following time step t, one computes $c_t := \exp(-0.5(\boldsymbol{Y}_t - \boldsymbol{Y}^{\text{old}})^T A_t(\boldsymbol{Y}_t - \boldsymbol{Y}^{\text{old}}))$, for a symmetric positive-definite matrix A_t, and triggers a new transmission if $U_t \sim U([0,1])$ realizes a value larger than c_t, making a new transmission more likely if the current observation strongly deviates from the previously transmitted observation. They assume the GSTM distribution from [152] for the state distribution $p(\boldsymbol{X}_t|\boldsymbol{X}_{t-1}, \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ represents the USV model parameter vector. They use VB in order to compute an approximate joint posterior for $\boldsymbol{X}_t$ and $\boldsymbol{\theta}$. They compare different adaptive and event-triggered UKF versions and the standard UKF with their method, which outperforms its competitors in terms of accuracy, both in a simulation as well as on a real-world experiment. As for an adaptive KF, the observation noise covariance parameters are stochastic, so a joint posterior for the distribution of the states and parameters must be found, which is done by an VB approximation in [274].

[368] consider multi-sensor fusion and propose to robustify the single filters by assuming t-distributed noise. The posterior state distribution for each filter is approximated by variational inference. Assuming that each sensor operates independently, they derive a weighting strategy which additionally neglects any dependence between the individual state components, resulting in a diagonal weight matrix which can be easily computed, as matrix inverses are avoided, for the price of potentially reduced accuracy. They compare their algorithm with competitor robust KF and sensor fusion algorithms in a real-world experiment with an autonomous driving platform, achieving better accuracy than its competitors. As for the computational time, their algorithm requires around 4 times more time than the standard KF, but around half of the time required for the federated KF.

[184] consider a leader-slave cooperative navigation setting where a fleet of slave vehicles with cheap and low-accuracy sensors is given, and one or multiple leader vehicles with high-accuracy sensors. As for the observations, they

consider the range between leaders and slaves. The states and measurements are the concatenated state and measurement vectors. In the resulting EKF, they model both the process noise and the measurement noises by t-distributions. The joint state and measurement posteriors are updated recursively via linearization of the state and the measurement equation. They apply their algorithm for underwater navigation in a real-world experiment with one slave and one leader vehicle. [151] consider t-distributed measurement and process distributions and use variational inference for approximating the posteriors. They apply their method for collaborative localization with two leader surface vehicles and one autonomous underwater vehicle.

[297] argue that in collaborative localization with low overlap between the local maps of the individual agents, outlier data associations are likely, resulting in potentially high outlier ratios. Therefore, they compute the spatial consistency between each two matched point pairs $\{\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)}\}$, $\{\boldsymbol{x}^{(2)}, \boldsymbol{y}^{(2)}\}$, considering them only as inliers if the difference $|||\boldsymbol{x}^{(1)} - \boldsymbol{x}^{(2)}||_2 - ||\boldsymbol{y}^{(1)} - \boldsymbol{y}^{(2)}|||$ is below some threshold. Local maps with more inliers are associated with a higher overall inlier probability in an EM algorithm where the positions are updated. Their algorithm is applied to a KITTI dataset and a real-world dataset with three robots.

[344] consider decentralized collaborative localization, which, in contrast to centralized collaborative localization where one central entity jointly estimates the states of all robots based on the transmitted data, allows the robots to share its own state estimates with each other. In addition to a robot-individual state equation, [344] define observation equations for the absolute range measurement $y_{a,t}^{il}$ between robot i and landmark l at time t, given by

$$y_{a,t}^{il} = h_t^i(\boldsymbol{X}_t^i, \boldsymbol{X}_t^l) + \boldsymbol{\nu}_t^i,$$

and for the relative range measurements $y_{r,t}^{ij}$ between robots i and j at time t, given by

$$y_{r,t}^{ij} = h_t^i(\boldsymbol{X}_t^i, \boldsymbol{X}_t^j) + \boldsymbol{\nu}_t^i.$$

While using Gaussian distributions for the state equations, [344] allow for t-distributed noise in the absolute and relative range measurements and propose a variational Bayes approach in order to update them for each robot. Due to shared information, interdependences between each robot pair have to be integrated into the algorithm, which themselves are updated iteratively. They apply their method for collaborative localization of 5 robots.

[200] propose to robustly estimate $V_t = \mathrm{Cov}(\boldsymbol{\epsilon}_t)$ of the KF by scaling the diagonal entries using different weight functions such as the IGGIII weight function. A similar idea has been proposed in [188] who scale the covariance matrix of the CKF using IGGIII weights. See also [294], [315], [318].

[352] combine an adaptive KF and a robust KF. In the adaptive KF, the gain and covariance are scaled with Huber weights. In the robust KF, IGGIII weights are used in order to scale the measurement covariance matrix. Finally, the estimated state vector and state covariance matrix is computed as a convex combination of both predictions from the adaptive KF and from the robust KF. Here, for small residuals, the adaptive KF gets more weight, and vice versa for large residuals. They apply their algorithm for land vehicle navigation.

[45] assume a linear state-space model propose to compute the standard Mahalanobis distance of the observations and to scale the noise covariance matrix V_t in the KF with a scalar if the Mahalanobis distance exceeds some threshold, where the scaling factor is chosen adaptively. They apply their algorithm to a kinematic positioning problem where both position and velocity have to estimated.

[304], [305] consider the observation equation $\boldsymbol{Y}_t = Z_t\boldsymbol{X}_t + \boldsymbol{\epsilon}_t + \boldsymbol{u}_t$, where $\boldsymbol{u}_t$ follows a non-ideal distribution. Note that this contamination scheme is not a convex contamination, unless one would consider the distribution of $\boldsymbol{\epsilon}_t + \boldsymbol{u}_t$ as contaminating distribution and the contamination radius as 1. In particular, they follow an unknown variance prior approach where $\boldsymbol{u}_t \sim \mathcal{N}(\boldsymbol{0}, \Sigma_t)$ where Σ_t itself follows some prior distribution. The estimation of Σ_t is done via the EM algorithm. They apply their approach for the localization of a marine vehicle and a quadrotor. The computation time was around 6 times higher than that for the standard KF, but lower than for the KF proposed in [2].

[231] consider object tracking by sequences of images and propose to apply a Kalman smoother for each pixel in order to deal with abrupt lightning changes and occlusions. They propose to replace the square in the Gaussian distribution by the Huber function, i.e., the observation model given some template feature vector $\boldsymbol{f}_t$ at time t is given by

$$p(\boldsymbol{Y}_t|\boldsymbol{f}_t) = c^{-1}|R|^{-1/2}\exp(-\rho_H(r(\boldsymbol{Y}_t, \boldsymbol{f}_t)))$$

for a normalizing constant c, a scale matrix R and the error

$$r(\boldsymbol{Y}_t, \boldsymbol{f}_t) = \sqrt{(\boldsymbol{Y}_t - \boldsymbol{f}_t)^T R^{-1}(\boldsymbol{Y}_t - \boldsymbol{f}_t)}.$$

They then approximate the posterior $\boldsymbol{f}_t$, which is no longer analytically computable. In their workflow, they first match templates. Given a set of predicted feature vectors $\hat{\boldsymbol{f}}_t(\boldsymbol{x})$, they match them to the current image in order to derive the errors for the KF at the next time step. Considering translation, rotation and scaling, they consider the transformation

$$T(\boldsymbol{\xi})(\boldsymbol{x}) = (1 + \xi_4)\begin{pmatrix}\cos(\xi_3) & -\sin(\xi_3)\\ \sin(\xi_3) & \cos(\xi_3)\end{pmatrix}\begin{pmatrix}x_1\\ x_3\end{pmatrix} + \begin{pmatrix}\xi_1\\ \xi_2\end{pmatrix}$$

with a parameter $\boldsymbol{\xi} \in \mathbb{R}^4$ to be estimated, which is done by robust regression w.r.t. the objective

$$\sum\nolimits_{\boldsymbol{x}} \rho_H(r(\boldsymbol{I}_t(\boldsymbol{p}(\boldsymbol{x}, \boldsymbol{\xi})), \hat{\boldsymbol{f}}_t(\boldsymbol{x})))$$

for the feature vector $\boldsymbol{I}_t(\boldsymbol{p}(\boldsymbol{x}, \boldsymbol{\xi}))$ observed at image point $\boldsymbol{p}(\boldsymbol{x}, \boldsymbol{\xi})$. The templates are updated using the robust KF and lastly, the scale matrix R is updated.

[334] apply a robust particle filter on pedestrian tracking using radar. They consider a non-linear state space model and assume $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0}, W_t^{-1}R)$ for a diagonal weight matrix $W_t = (w_{t,m})_{m=1,\ldots,q}$, with Gamma priors

$$w_{t,m} \sim \text{Gamma}(w_{t,m}|\theta_m/2, \theta_m/2),$$

$m = 1, ..., q$.

[29], [30] consider joint robust GNSS position and attitude estimation with the EKF. As the EKF can be interpreted as the optimization problem

$$\hat{\boldsymbol{X}}_{t|t} = \text{argmin}_{\boldsymbol{x}}(||\boldsymbol{x} - \hat{\boldsymbol{X}}_{t|t-1}||^2_{\Sigma_{t|t-1}} + ||h(\boldsymbol{x}) - \boldsymbol{Y}_t||^2_{V_t}),$$

they point out that GM-estimators such as in [97] do not allow for redescending losses, hence they use robust information filters where the optimization problem is re-formulated by replacing V_t by a weighted version $V_t^{1/2} W^{-1} V_t^{T/2}$ for

$$W = \text{diag}(w(V_t^{-1/2}(\boldsymbol{Y}_k - h(\boldsymbol{X}_k))))$$

for a weight function w, which may be the Huber loss function or the Tukey loss function. The solution to the optimization problem above can then be approximated iteratively. They further adapt this strategy to the situation where the data points belong to a manifold, as in the joint position and attitude estimation problem. Experiments are done on simulated data with an outlier rate of 20% and 25% in [30] and [29], respectively.

Apart from approaches based on a robustification of the loss function or modeling heavy-tailed noise, there are strategies that invoke other techniques for robustification, such as **outlier detection or clipping**.

If only AOs occur, [267] propose a Huberization of the Kalman gain in order to robustify the correction step, i.e., the Huberization of the Kalman gain is given by

$$H_b(K_t \boldsymbol{\Delta Y}_t) := K_t \boldsymbol{\Delta Y}_t \min\left(1, \frac{b}{|K_t \boldsymbol{\Delta Y}_t|}\right).$$

The clipping height b may be determined so that a certain Anscombe-efficiency level is attained or by a minimax criterion w.r.t. a least favorable contamination radius. In the case of SO-outliers, they assume a convex contamination model around the true distribution $F^{\text{id}}_{\boldsymbol{Y}_t}$ of the $\boldsymbol{Y}_t$, leading to a distribution $F^{\text{re}}_{\boldsymbol{Y}_t}$, so that, assuming independence with the distribution $F_{\boldsymbol{X}_t}$ of the $\boldsymbol{X}_t$, the ball

$$\mathcal{U}^{\text{SO}}(r) = \bigcup_{0 \le s \le r} \{\mathcal{L}(\boldsymbol{X}_t, \boldsymbol{Y}^{\text{re}}_t) \mid F^{\text{re}}_{\boldsymbol{Y}_t} \in U_c(F^{\text{id}}_{\boldsymbol{Y}_t}, s)\}$$

is considered. They propose to either minimize the MSE on $\mathcal{U}^{\text{SO}}$ or to minimize the MSE w.r.t. a bound on the bias on $\mathcal{U}^{\text{SO}}$. In the case of IOs, they show that the correction step can be written as

$$\boldsymbol{X}_{t|t} = \boldsymbol{X}_{t|t-1} + Z^{\Sigma}_t(\boldsymbol{\Delta Y}_t - \mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\Delta Y}_t]), \quad Z^{\Sigma}_t = \Sigma_{t|t-1} Z_t^T (Z_t^T \Sigma_{t|t-1} Z_t)^-,$$

so a robustification is done by Huberizing $\mathbb{E}[\boldsymbol{\epsilon}_t|\boldsymbol{\Delta Y}_t]$, which equals $(I - Z_t K_t)\boldsymbol{\Delta Y}_t$ in the ideal model, leading to the Huberized correction step

$$\boldsymbol{X}_{t|t} = \boldsymbol{X}_{t|t-1} + Z^{\Sigma}_t[\boldsymbol{\Delta Y}_t - H_b((I - Z_t K_t)\boldsymbol{\Delta Y}_t)].$$

In [269], the same clipping strategy is performed for the EKF. They apply their method to vehicle tracking with the goal to estimate the change of altitude. Apart from measurement errors and changes in the road surface, their data also consist of missings due to signal loss, e.g., in tunnels, leading to jumps in altitude or speed.

[359] propose an iterated EKF where the linearization is considered at the updated state $\boldsymbol{X}_{t+1|t}$. Then, a Huberization of the Kalman gain $K_t \boldsymbol{\Delta Y}_t$, going back to [267], is performed. They consider spacecraft navigation.

[91] point out that robust KFs that allow for heavy-tailed noise distributions and that approximate the posterior by variational Bayes are only robust to additive outliers, while a Huberization of the residuals such as done in [269] also robustifies against IOs. Moreover, they point out that anomalies are often multi-modal, which cannot be represented by t-type distributions.

[38] use a standard KF, but reduce the data set to inliers using RANSAC before, for lane detection and tracking.

[37] consider measurement outliers in collaborative localization and downweight them in an KF scheme where the weights are computed via the Stahel-Donoho estimator.

[196] consider additive outliers and propose to identify them by computing the matrix

$$B_t = Z_t P_{t|t-1} Z_t^T + Z_t Q_t Z_t^T + V_t.$$

The i-th component of an observation is flagged as outlier if the i-th diagonal element of $(\boldsymbol{Y}_t - Z_t \boldsymbol{X}_{t|t-1})(\boldsymbol{Y}_t - Z_t \boldsymbol{X}_{t|t-1})^T$ is larger than B_{ii}, multiplied with some weight. If an outlier is detected, the corresponding predicted state and covariance matrix are corrected via component-wise scaling. Their method is applied to aircraft tracking. They argue that their algorithm achieves real-time performance as the computation time is lower than 100 ms per frame.

[380] integrate outlier detection and suppression into a variational Bayes approach by using sliding windows. First, they allow for heavy tails by using a t-distribution for both the measurement and process noise. In their sliding window approach, the posterior for each time step in the window are updated by using a constant measurement and state covariance matrix within each sliding window. In addition, in each window, for each covariance matrix, an auxiliary variable is considered which scales the covariance matrix, allowing for outlier suppression in the respective window. The posterior for the states, covariance matrices and auxiliary variables is approximated jointly by variational inference. They apply their method to tracking a car in a simulated and in a real-world experiment.

[87] point out that the EKF performs linear approximations based on the estimations from the previous step, so that errors may even increase. They also point out that the computational complexity dramatically increases when trying to robustify the EKF. They criticize the $\mathcal{H}_\infty$-filtering approach (e.g., [285]; see Sec. V), which interprets outliers as bounded uncertainty, to be too pessimistic. The idea in [87] is to detect outliers in the innovations and to clip them, i.e.,

$$\max(-k, \min(k, Y_{i,t} - (h_i(\hat{\boldsymbol{X}}_{t|t-1}))),$$

where the clipping height k is adaptively chosen.

B. ESTIMATION OF VEHICLE PARAMETERS

KFs intend to provide an estimation of the true underlying state $\boldsymbol{X}_t$, based on the observed noisy state $\boldsymbol{Y}_t$. Of course, provided that an underlying state space model can be formulated, one can estimate vehicle parameters by considering them to be part of the underlying states $\boldsymbol{X}_t$.

A Huberized linear KF has been applied in [83] for position estimation of vehicles, in [194] who estimate the position error and mounting angles and yaws as well as lever-arm residuals between data from the INS and a laser Doppler velocimeter or VO, respectively.

A Huberized EKF is applied in [332] for center of gravity estimation, which is done by using a state space model that relates the height of the center of gravity and its distance to the front axis with the velocity, [369] consider estimating the rotor angle and speed of a bus. [68] apply robust KFs, including a Huberized EKF and a covariance-scaled EKF, for the estimation of attitude, position and velocity errors, and acceleration and gyro biases of a rover. Several robust KFs, including a Huberized KF, the KF from [45] and several variational filters, have been compared in [69], here with the application to improve wheel-inertial odometry for planetary rovers. The state represents the attitude, position and velocity error, and the acceleration and gyro biases.

[292] use a Huberized UKF with adaptive covariance for the navigation of coupled vehicles, [313] consider vehicle state estimation, such as longitudinal and lateral velocities, yaw rate, mass, center of gravity, and moment of inertia.

[190] use a Huberized CKF for rotor angle and speed estimation and confirm real-time performance as their algorithm only leads to a slight overhead in the computation time compared to the standard CKF. [189] apply a Huberized CKF to an INS where the state consists of attitude, latitude, longitude, height and velocity errors, gyroscopic drifts, accelerometer biases, and a scale factor error of the Doppler velocity log. A numerically more stable version of the CKF, the square-root CKF, has been Huberized in [139], who use it for state-of-charge (SoC) estimation for lithium-ion batteries. Here, the state represents the SoC and the polarization voltage.

The variational filter from [2] has been applied in [277] for the estimation of the internal resistance in the battery system of electric vehicles. [357] apply an EKF with t-distributed observation noise for ship position estimation. [127] use an adaptive KF with the MCC and allow for a time-varying noise covariance via variational Bayes in order to estimate the tire-road forces and the sideslip angle of a vehicle. In numerical simulations, their algorithm only requires slightly more computation time than the standard CKF.

[52] use the correntropy criterion for the linear KF for the navigation of vehicles in an urban environment. [61] combine the CKF with maximum correntropy for spacecraft attitude estimation, [248] consider the estimation of the yaw rate, the lateral and the longitudinal velocity of a vehicle, [50] consider car mass estimation. [103] use the MCC for the square-root CKF in order to estimate velocities, the yaw rate, and wheel rotation of electric vehicles, [201] consider estimating the yaw rate, side slip angle and the longitudinal velocity.

[249] scale the covariance matrix corresponding to the measurement noise of the KF via IGGIII weights and estimate the SoC of lithium-ion batteries of electric vehicles.

[289] consider, under a static environment assumption, a linear relation between Doppler velocities (based on radar measurements) and the ego-velocity. Due to dynamic features, outliers are generated, which are first filtered using a sliding window approach where instances where the velocity of two subsequent measurements differs too much or where the velocity differs too much from the average in the window are discarded. With the filtered data, robust regression using truncated least squares and the Cauchy loss is performed.

Robust regression can also be used for velocity estimation, see, e.g., [300], who consider the objective

$$\min_{\boldsymbol{\beta}} \left(\sum\nolimits_i \rho_H(v_i - \boldsymbol{X}_i \boldsymbol{\beta}) \right),$$

for the measured velocities v_i and variables such as the mean traffic speed or the road curvature, that are represented by the $\boldsymbol{X}_i$.

[258] apply robust M-estimators such as LTS and LMS, but also consider robustifications of recursive least squares, for vehicle parameter estimation. Such recursive objective are important when considering time-varying systems so that for each time step t, a solution can be efficiently computed, which is in particular important for real-time applications. The objective is

$$\min_{\boldsymbol{\beta}} \left(\sum\nolimits_{t=t_0}^{T} \lambda^{t-t_0} \rho(Y_t - \boldsymbol{X}_t \boldsymbol{\beta}) \right)$$

for some forgetting parameter $\lambda \in]0, 1]$. [258] also propose to additionally regularize recursive least squares. They apply these algorithms to mass estimation and tractive force prediction of vehicles in grey-box models. They use the model $\boldsymbol{X}\boldsymbol{\beta} = Y$ for

$$\boldsymbol{X} = \left(g\cos(\theta), g\sin(\theta) + v', v^2, \frac{v^4}{4r_P^2} \right), \quad \boldsymbol{\beta} = \left(m f_{r_0}, m, \frac{\rho_a}{2} A c_x, \frac{m^2}{c_{\mathrm{yW}}} \right),$$

for the gravitational constant, g, the gradient angle, θ, the path radius, r_P, the rolling resistance coefficient, f_{r_0}, the mass, m, the vehicle cross-sectional area, A, the longitudinal drag coefficient, c_x, the air density, ρ_a, the wheel-concerning stiffness, c_{yW}, and the velocity, v. $\boldsymbol{X}$ takes the role of the measured input and Y represents the tractive force, which takes the role of the measured output in mass estimation. Note that a robust estimation of $\boldsymbol{\beta}$ allows for extracting the desired vehicle parameters, here the mass, from the estimate $\hat{\boldsymbol{\beta}}$. In [258], Y itself is computed by the model

$$Y = \frac{T_R - I_W \theta''_W}{r_W},$$

for the rim torque, T_R, the wheel moment of inertia, I_W, the dynamic wheel radius, r_W, and the second derivative, θ''_W, of the wheel rotation angle, θ_W. They also provide an overview of variables that need to be estimated as they cannot be directly measured. In the above model, the engine torque and the reduced moment of inertia can be assessed by look-up tables, while the velocity, the path angle, and the path radius can be estimated by simple models. They also point out that in the presence of outliers, the MSE is not the correct criterion for validation.

Mass estimation has also been considered in [60]. Under the assumption of a nearly flat road, the model $Y = \boldsymbol{X}\boldsymbol{\beta}$ for the longitudinal acceleration Y is approximately valid, where

$$\boldsymbol{X} = \left(\frac{T_e i_g i_f \eta_T}{r_W} - \frac{c_x A \rho_a v^2}{2}, -g \right), \quad \boldsymbol{\beta} = \left(\frac{1}{m}, f_{r_0} + \frac{F_{err}}{mg} \right)^T,$$

where T_e is the engine torque, i_g the transmission gear, i_f the final drive ratio, η_T the driveline mechanical efficiency, and where F_{err} represents the error of a physical model of the driving force. They consider recursive regression where the objective consists of two parts, one for each component of $\boldsymbol{\beta}$. The squared loss is replaced by the three-part redescender going back to [123] for the first part of the objective. In their simulations, they confirm real-time capabilities of their algorithm.

[67] consider the estimation of running resistances of a train. To this end, they invoke the differential equation

$$\partial_t v(t) = u(t) - r(v(t)) - w(s(t)), \quad w(s(t)) = \frac{g}{\rho_m} p(s(t)) + \frac{k}{\rho_m r_T(s(t))},$$

where $u(t)$ is the tractive and brake effort, $r(v(t))$ the running resistance corresponding to the velocity, at time t, respectively, and where ρ_m is the rotational mass factor, $p(s(t))$ the gradient of the track at $s(t)$, $r_T(s(t))$ is the radius of the track at $s(t)$, and where k is a gauge factor corresponding to the impact of a curve on the train. For the resistance, they use the model

$$r(v(t)) = r_0 + r_1 v(t) + r_2 v^2(t),$$

for rolling resistance parameters r_0, r_1, r_2. Using time discretization with time step size Δt, one gets the regression model $Y = \boldsymbol{X}\boldsymbol{\beta}$, for

$$y_k = u_k - w(s_k) - a_k, \quad a_k = \frac{v(t_{k+1}) - v(t_k)}{\Delta t},$$
$$\boldsymbol{\beta} = (r_0, r_1, r_2)^T, \quad \boldsymbol{X}_k = (1, v_k, v_k^2),$$

with the respective quantities at time step k. They consider a plethora of non-robust and robust loss functions, including the Huber, Tukey, Cauchy, and Welsch loss.

[32] consider robust parameter estimation for electric vehicles, including mass, braking parameters, drag and resistance, electric parameters of asynchronous machines (such as resistances, inductivities), and parameters of lithium-ion cells (voltage, SoC, State of Health (SoH)). They point out that contamination may arise from wrong measurements, disturbed transfer, phases with low system stimulation (when driving with constant speed), wrongly modelled system dynamics, or wrong input parameters such as wheel radii, velocity, driving torque, air density, or acceleration. As for the longitudinal vehicle dynamics, they consider the model $Y = F_A = \boldsymbol{X}\boldsymbol{\beta}$ for

$$\boldsymbol{X} = \left(a_x, g, g v_x, g v_x^4, 0.5 \rho_a v_x^2, a_y^2\right),$$
$$\boldsymbol{\beta} = \left((m, mc_{r,0}, mc_{r,1}, mc_{r,4}, c_w A_{F_{z_g}}, \frac{m^2}{(4c_{\gamma_{\text{Rad}}})^T} \right),$$

for the longitudinal velocity, v_x, the longitudinal acceleration, a_x, the lateral acceleration, a_y, the rolling drag forces, $c_{r,j}$, the air drag force, c_w, the projected surface area projected on the y-z-axis, $A_{F_{z_g}}$, and the curve drag force, $m^2/(4c_{\gamma_{Rad}})$. Apart from outlier detection, which has disadvantages when being applied to embedded systems due to a large memory and computational burden due to the recursive estimations, [32] propose a robust version of recursive least squares, including exponential forgetting, regularization and parameter range constraints, see [32, Alg. 3.8], which is solved by an IRWLS procedure. As for the robustness aspect, the weighted RLS objective

$$\min_{\boldsymbol{\beta}} \left(\frac{1}{n} \sum\nolimits_{t=t_0}^{T} \lambda^{t-t_0} w(r_t^2(\boldsymbol{\beta})) \right)$$

is considered, where w is a weight function such as the Huber weight function. As for electric parameters, they consider the model

$$Y = \partial_t u_S^\alpha + n_p \omega_m u_S^\gamma = \begin{pmatrix} \begin{pmatrix} \partial_t^2 i_S^\alpha + \partial_t (n_p \omega_m i_S^\gamma) \\ \partial_t i_S^\alpha \\ -u_S^\alpha \\ \partial_t i_S^\alpha + n_p \omega_m i_S^\gamma \\ i_S^\alpha \end{pmatrix} \end{pmatrix}^T \begin{pmatrix} \sigma_{Bl} L_S \\ L_S \frac{R_R}{L_R} \\ \frac{R_R}{L_R} \\ R_S \\ R_S \frac{R_R}{L_R} \end{pmatrix} = \boldsymbol{X}\boldsymbol{\beta},$$

for currents, i, voltages, u, electric resistances, R, and inductivities, L, where the subscripts S and R refer to the stator and the rotor, respectively, and where superscripts α and γ refer to the coordinate system of the asynchronous machine. ω_m is the mechanical rotor drive, n_p the number of pole pairs, and σ_{Bl} the scattering coefficient. Although they do not apply their robust RLS procedure here, they analyze problems that arise when applying the non-robust variant and propose a Savitzy-Golay smoothing of the signal in order to compute $\partial_t^2 i_S^\alpha$. In lithium-ion cells, the model

$$Y^{(k)} = U_{\text{akk}}^{(k)} = (1, U_{\text{akk}}^{(k-1)}, I_{\text{akk}}^{(k)}, I_{\text{akk}}^{(k-1)}) \boldsymbol{\beta}$$

for $\boldsymbol{\beta} = ((1 - a_1 U_{\text{OC}}, a_1, a_2, a_3)$ is assumed, for voltage U_{akk}, current I_{akk} and open circuit voltage U_{OC} of the cell, and quantities a_1, a_2, a_3 that are given in terms of the time step size, inner resistance R_0, and resistance R_1 and capacity

C_1 of the RC branch, respectively. They apply their robust RLS variant here.

C. BOUNDING BOX ESTIMATION

[154] propose a robust estimation of future bounding boxes, including their uncertainty. Given an anchor box $\boldsymbol{B}_0$, let $T(t) = [T_x(t), T_y(t), T_w(t), T_h(t)] : \mathbb{R} \to \mathbb{R}^4$ be the transformation at time step t from $\boldsymbol{B}_0$ to the ground-truth prediction box $\boldsymbol{B}^*(t)$ where the indices represent the x- and y-position of the center, the width and the height of the bounding box, respectively. Then, the proposed confidence-weighted Huber loss is

$$H_k(u, u', \sigma) = \ln(c) + \begin{cases} (u-u')^2/(2\sigma^2), & |u-u'| < k \\ k\sigma^{-2}|u-u'| - k^2/(2\sigma^2), & |u-u'| \geq k \end{cases},$$

for a normalizing constant c and a scale parameter $\sigma > 0$. They propose to set k to the estimated uncertainty $\hat{\sigma}$, scaled by some constant factor. As for the objective, they consider minimizing the discrepancies of the dimension-individual means, i.e.,

$$\min_{\boldsymbol{\theta}_B^d, \boldsymbol{\theta}_\sigma^d} \left(\sum_d H_d(T_d^*(t), \hat{T}_d(t, \boldsymbol{\theta}_B^d), \hat{\sigma}_d(t, \boldsymbol{\theta}_\sigma^d)) \right),$$

where $\hat{T}_d$ and $\hat{\sigma}_d$ are estimators for the transformation and the uncertainty by a neural network, with individual parameters $\boldsymbol{\theta}_B^d$ and $\boldsymbol{\theta}_\sigma^d$, respectively. Experiments were performed on the KITTI "raw" dataset. The first 20 frames of a tracklet serve as training sample, the prediction horizon consists of the following 10 frames.

The Huber loss is also referred to as "smooth l_1-loss", only up to a scaling, in the deep learning literature. For example, [53] use this loss function as orientation loss and bounding box offset loss, [54] for the 3D box regression loss, [175] use it for all regression losses in 3D object detection, or in 2D object detection algorithms such as Fast R-CNN [108]. [15] propose to use a convex combination of the IoU loss and the Huber loss.

D. DETECTION OF ROAD FEATURES

[302] consider the detector YARF (yet another road follower), which uses Robust Statistics in order to detect road features. They propose the model

$$Y_i = \beta_0 - \beta_0^r + \beta_1 X_i + 0.5\beta_2 X_i^2,$$

where β_0^r is the offset from the road spine, β_0 the Y-intercept of the spine arc, β_1 the heading w.r.t. the tangent of the spine arc at β_0 and where β_2 is the curvature of the spine arc, where the positions are given by $(\boldsymbol{X}_i, Y_i)$. The coefficients are estimated using LMS.

[234], [235] consider the problem of road surface extraction from 3D point clouds and apply a robust variant of locally weighted regression based on the Tukey loss, i.e., they minimize

$$\sum_i \rho_T(\tilde{r}_i) w(\boldsymbol{X}_i)(Y_i - f_{\boldsymbol{\beta}}(\boldsymbol{X}_i))^2,$$

where $\tilde{r}_i = r_i/\hat{\sigma}$ for $\hat{\sigma}$ being the MAD of $(|r_1|, ..., |r_n|)$, where the weight function w is the tri-cube weight function

$$w(\boldsymbol{X}_i) = \begin{cases} \left[1 - \left(\frac{||\boldsymbol{X}_i - \boldsymbol{X}_j||_2}{\max_{j \in N(\boldsymbol{X}_i)}(||\boldsymbol{X}_i - \boldsymbol{X}_j||_2)}\right)^3\right]^3, & j \in N(x) \\ 0, & j \notin N(x) \end{cases},$$

for a local neighborhood $N(\boldsymbol{X}_i)$ of $\boldsymbol{X}_i$, and for some potentially non-linear function $f_{\boldsymbol{\beta}}$. The residuals after the fit indicate whether the individual points belong to the road surface or whether they are non-ground/3D surface points.

E. OTHER APPROACHES

Outlier detection is a popular topic in data analysis. It is therefore out of scope for this paper to list all the literature where some kind of outlier detection has been performed in the context of autonomous driving. Just as an example, consider the work of [345] who have data from n microphone arrays which are located as known 2D-positions $\boldsymbol{p}_i$, $i = 1, ..., n$, which measure angles of arrival from an object with the goal to determine its position $\boldsymbol{p}_0$. In an iterative manner, first the m microphone arrays with the smallest distance to a particular object are identified based on an initial estimate $\hat{\boldsymbol{p}}_0$ of the object's position, which form a reference set. Then, the matrices $P_i = (\hat{\boldsymbol{p}}_{ij})_{j=m+1,...,n}$, $i = 1, ..., m$, are formed, where $\hat{\boldsymbol{p}}_{ij}$ is the estimated position of the object when replacing $\boldsymbol{p}_j$ with $\boldsymbol{p}_i$. Using the robust Mahalanobis distance where the mean and covariance of the P_i are estimated robustly in the spirit of the Gnanadesikan-Kettenring estimator [109], some instances are flagged as "outliers". Finally, the position of the object is estimated using weighted least squares where the outlying instances are downweighted accordingly.

[36] use median regression for trend estimation in GPS time series based on the Theil-Sen estimator

$$\hat{v} = \text{med}_{i<j}\left(\frac{z_j - z_i}{t_j - t_i}\right)$$

for the velocity, where each z_i represents the coordinate at time t_i. As a pre-processing step, outlier detection is done where the slope is computed for each data pairs, removing all pairs for which the slope has a distance larger than 2 MADs from the median.

[204] estimate the orientation changes of a vehicle based on radar images. After an MAD-based outlier removal, the surviving pairs of reference and data images are considered and the rotation and scale is estimated using Tukey's biweight function. At the end, the estimation is refined by minimizing the Cauchy loss, evaluated at Mahalanobis-type residuals arising from the previous estimation step.

[23] propose a whole family of loss functions, including robust ones as special cases, given by

$$\rho(r, \alpha, \tau) := \frac{|\alpha - 2|}{\alpha}\left(\left(\frac{\left(\frac{r}{\tau}\right)^2}{|\alpha - 2|} + 1\right)^{\alpha/2} - 1\right),$$

with a scale parameter $\tau > 0$ and a shape parameter α. The special cases $\alpha = -\infty$, $\alpha = -2$, $\alpha = 0$, $\alpha = 2$ correspond to the Welsch loss, the Geman-McClure loss, a smoothed version of the l_1-loss and the squared loss, respectively. They apply several particular loss functions from this family to tasks such as monocular depth estimation and fast global registration.

[347] consider the truncated least squares (TLS) problem

$$\min_z \left(\sum_i \min \left(\frac{(z - Y_i)^2}{\sigma_i^2}, \tau^2 \right) \right)$$

for the inlier standard deviation σ_i corresponding to Y_i. They show that general geometric perception problems such as pose, rotation or 3D structure estimation can be formulated as TLS problem. They solve it by a convex relaxation. Their relaxation is extended in [348] to robust loss functions such as the Huber loss or Tukey's biweight loss where it is applied to, for example, point cloud registration, pose estimation, shape estimation, and rotation averaging.

[241] propose the version

$$\rho_{H,k}(Y_i, \hat{Y}_i) = \begin{cases} |\hat{Y}_i - Y_i|, & |\hat{Y}_i - Y_i| \leq k \\ \frac{(\hat{Y}_i - Y_i)^2 + k^2}{2k}, & |\hat{Y}_i - Y_i| \geq k \end{cases}$$

of the Huber loss function for depth estimation, where the Y_i are the pixel values in the ground truth depth-map, the $\hat{Y}_i$ their predictions, and where the threshold k is given by $k = 0.2 \max_i(|\hat{Y}_i - Y_i|)$. This loss is one component of an overall loss that is composed by this pixel loss and a loss function for structural similarity and for the intensity gradients of the pixels, respectively. Contamination may arise from inherent blurs in images.

As for graphical neural networks (GNNs), which are a backbone of many computer vision methods, [105] consider a robust aggregation of the embedded features of neighboring points in GNNs. They replace the usual sum or mean aggregation, which opens the door for distorted aggregated embeddings due to single perturbed points (as the BDP of an arithmetic mean is zero), by a smoothed medoid aggregation, which is computed by

$$\sum_i w_i \boldsymbol{X}_i, \quad w_i = \frac{\exp\left(-\delta^{-1} \sum_j ||\boldsymbol{X}_j - \boldsymbol{X}_i||\right)}{\sum_k \exp\left(-\delta^{-1} \sum_j ||\boldsymbol{X}_j - \boldsymbol{X}_k||\right)}$$

for some parameter δ that controls the approximation to the original solution

$$\text{argmin}_{\boldsymbol{y}} \left(\sum_i ||\boldsymbol{X}_i - \boldsymbol{y}|| \right).$$

The solution to the smoothed medoid problem approaches the arithmetic mean for $\delta \to \infty$ and the exact medoid for $\delta \to 0$. They show that the soft medoid procedure has a BDP of 0.5. As for contamination, they assume that an adversary can perturb a fraction of the aggregation points. [104] propose the Soft Median aggregation which requires less memory capacities than the Soft Medoid aggregation while maintaining the BDP of 0.5. It is given by

$$\text{softmax}(-\boldsymbol{c}\delta^{-1} p^{-1/2})^T X$$

for the vector $\boldsymbol{c}$ consisting of components $c_j = ||\bar{\boldsymbol{X}} - \boldsymbol{X}_j||$ and for the node attributes $X \in \mathbb{R}^{n \times p}$ of the graph to which the GNN is applied.

Deep fundamental matrix estimation has applications in 3D perception, for example, for the projected retinal image coordinates $\boldsymbol{p}$ corresponding to 3D coordinates of the corresponding point, a fundamental matrix F satisfies $\boldsymbol{p}^T F \boldsymbol{p} = 0$. Having an initial estimate for F, [367] propose to refine F by computing the signed distances r_i^2 and flagging all points for which r_i^2 is larger than a certain multiple of a robust scale estimate as outliers and then consider an LTS approach where only the residuals from the non-flagged instances enter. [254] propose to estimate the inlier distribution during the optimization. In their context, the optimization problem is

$$\min_{\boldsymbol{\theta}} \left(\sum_i ||A \boldsymbol{p}_i \boldsymbol{\theta}||^2 \right) \quad \text{s.t.} \quad ||\boldsymbol{\theta}|| = 1,$$

for some matrix A. This problem can approximately be solved by solving iteratively

$$\boldsymbol{x}^{j+1} = \text{argmin}_{||\boldsymbol{x}||=1}(||W^j(\boldsymbol{\theta}) A \boldsymbol{x}||^2)$$

for a weight matrix $W(\boldsymbol{\theta})$. They propose to learn the weights by a deep neural network, so that they essentially have a meta-algorithm of IRWLS. Identifying the solution in one step of the IRWLS problem as a right singular vector of $W(\boldsymbol{\theta})A$ for the weight matrix $W(\boldsymbol{\theta})$, they show that $\boldsymbol{\theta}$ can be learned by backpropagation through an SVD layer. This technique is applied to fundamental matrix estimation.

V. APPLICATIONS IN AUTONOMOUS DRIVING: PREDICTION AND PLANNING

In this section, we collect robust approaches for prediction and planning. The first two subsections are devoted to reinforcement learning and imitation learning. Here, the ego-vehicle has to learn by experience (typically, via simulations) how to behave in which situations, so the own actions and, implicitly, the evolution of the states of surrounding vehicles, are learned. The third subsection considers model-predictive control, where models for vehicle dynamics are used in order to predict the state evolution for the surrounding traffic participants and the ego-vehicle based on the observed current state. In all cases, reliable perception is important (see Fig. 1). Nevertheless, robust planning approaches may safeguard partially against misperception, and can, in addition, also cope with other types of peculiarities such as adversarial driving behaviors of the surrounding traffic participants. The fourth subsection addresses Byzantine robustness, which becomes important when performing federated or distributed RL training.

A. REINFORCEMENT LEARNING

Reinforcement learning (RL) considers a Markov decision process $(\mathcal{S}, \mathcal{A}, \beth, \gamma, R)$ for a state space $\mathcal{S}$, an action space $\mathcal{A}$, a set $\beth$ of transitions, a discount factor $\gamma \in]0, 1]$, and a

reward function $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ that assigns a real-valued reward to a state-action-state triple (s_t, a_t, s_{t+1}), where s_{t+1} is the state into which the system is translated in the next time step after action a_t was executed in state s_t. The transition model may either be deterministic, so that $\beth \in \daleth$ is a mapping $(s,a) \mapsto \beth(s,a) \in \mathcal{S}$, or stochastic, so that a density value $P_\beth(s'|s,a)$ is assigned to (s,a). The goal is to learn a policy π which is either deterministic, i.e., a map $\pi : \mathcal{S} \to \mathcal{A}$ so that $\pi(s)$ is the action taken when being in state s, or stochastic, i.e., $\pi : \mathcal{S} \times \mathcal{A} \to [0, \infty[$ so that $\pi(s,a)$ assigns a density value to the state-action pair (s,a). We abbreviate $R(s_t, a_t, s_{t+1}) =: R_t$. The value function for a given state s is the expected future reward that the agent receives when following policy π, i.e.,

$$V^\pi(s) = \mathbb{E}_\pi \left[\sum_{t'=t}^{\infty} \gamma^{t'-t} R_t \middle| s_t = s \right].$$

The Q-function similarly assigns a value to a state-action pair (s,a) in the sense that, starting with $s_t = s$, one does not let the policy choose the initial action a_t but starts with a given action a at the initial state s.

In RL or MPC, the term "robustness" is often understood as adversarial robustness, hence many robust RL algorithms perform adversarial training. However, in contrast to adversarial attacks in classical machine learning where a model is trained on a static data set and where adversarial attacks are computed after model training, decoupling them from the training procedure, adversarial attacks in RL are used for adversarial training where an adversarial agent challenges the ego-agent, indeed affecting the training procedure. This makes the notion of adversarial robustness in RL inherently close to Robust Statistics.

For example, adversarial RL approaches have been applied to train an agent for autonomous driving, see [256], who propose a minimax game where the adversarial agent minimizes the objective that the ego agent aims at maximizing (up to a different regularization parameter). [247], [257] suggest similar minimax games. [256], [257] apply their method to a scenario where the ego vehicle aims at crossing a 4-way intersection, where adversarial vehicles drive on the lanes the ego vehicle has to cross. [134] consider adversarial attacks against the agent's observations in highway scenarios, which is trained according to maximizing the Jensen-Shannon divergence between the $\pi(s,\cdot)$ and $\pi(\tilde{s},\cdot)$, for the perturbed state $\tilde{s}$ of s. In [133], the worst-case observational perturbations are computed by an adversary using the FGSM scheme. They consider highway, intersection, and on-ramp scenarios with episode lengths of 300, 30 and 30 time steps, respectively. They observe much higher computational costs than for standard RL algorithms, which is a consequence of the approximation of the worst-case perturbations by a Bayesian approach. In [130], the management of a fleet of electric vehicles is considered, where one has one agent for each region of the map who can displace vehicles into adjacent regions or to charging stations. Here, the adversarial agent is a perturbation of the observed states of the region's agents, which consist of the number of vacant and low-battery vehicles, information about charging spots, and demand. The objective is a minimax game, and both the region's and the adversary's policies are updated iteratively. [84] propose to discount the adversary's reward and to constrain the number of attacks by an upper bound, encouraging to only attack in critical situations. Their method is applied for left-turn in an intersection and on-ramp merging. [117] consider state attacks and optimize the worst-case discounted reward, while the adversarial agents aim at performing the action that minimizes the victim's reward. In a simulation framework, they consider driving scenarios with an obstacle.

In addition, many approaches concerning robust control are given in the literature. One can distinguish between minimax games where one searches for the policy the maximizes the cumulated reward/minimizes the cumulated cost given the worst-case trajectory or worst-case transition model (similarly as above in the adversarial RL approaches, but with the difference that the adversarial agents trained in these approaches do not necessarily reflect worst-case situations), risk-sensitive criteria where an individual risk measure is used as objective, and constrained criteria where the reward should be maximized subject to several constraints, see [99].

More formally, the worst-case criterion under parameter uncertainty corresponds to the objective

$$\max_{\pi \in \Pi} \left(\min_{P \in \daleth} \left(\mathbb{E}_{\pi,P} \left[\sum_{t=0}^{\infty} \gamma^t R_t \right] \right) \right),$$

for a set $\daleth$ of transition matrices, and a set Π of policies, while the worst-case criterion under inherent uncertainty is given by

$$\max_{\pi \in \Pi} \left(\min_{\omega \in \Omega^\pi} \left(\mathbb{E}_{\pi,\omega} \left[\sum_{t=0}^{\infty} \gamma^t R_t \right] \right) \right),$$

for a set Ω^π of trajectories that are allowed under policy π [101]. E.g., [232] consider the worst-case criterion under parameter uncertainty for finite state-action spaces and apply the method to aircraft routing.

Another example of a minimax criterion is the robust Bellman TD operator introduced in [279]. The objective is the expected squared temporal difference

$$\frac{1}{2} \mathbb{E}_{o_t \sim P_O} [(\delta^{\text{nominal}}_{t,w'}(w_t, o_t))^2]$$

for

$$\delta^{\text{nominal}}_{t,w'}(w_t, o_t) = Y^{\text{nominal}}_{t,w'}(o_t) - Q_{w_t}(s_t, a_t)$$

for the weights w' of the target network, the current parameter w_t of the approximation $Q_{w_t}(s,a)$ of the true Q-function, the distribution P_O of the observations, and the nominal targets

$$Y^{\text{nominal}}_{t,w'}(o_t) = R_t + \gamma \max_{a'}(Q_{w'}(s_{t+1}, a'))$$

for observation $o_t = (s_t, a_t, R_t, s_{t+1})$ at time t. In the robust Bellman TD formulation, one still considers the squared TD error but a robust minimax target label

$$Y^{\text{robust}}_{t,w'}(o_t) = R_t + \gamma \min_{\mathrm{J}\in\mathcal{U}} \left(\sum\nolimits_{s'\in\mathcal{S}(s_t,a_t)} \mathrm{J}(s'|s_t,a_t) \max_{a'}(Q_{w'}(s',a')) \right).$$

for the set $\mathcal{S}(s_t,a_t)$ of all possible states at time $(t+1)$ given s_t and a_t under the uncertainty set $\mathcal{U}$.

[44] consider policy gradient descent and replace the squared differences of the predicted Q-values for given state-action pairs and the observations by the absolute differences and the Huber losses. They apply their algorithm for autonomous parking and consider scenarios with 100 time steps. In their experiments, the Huber loss allows for quicker convergence, finally resulting in an even decreased training time in comparison with training according to the MSE and the MAE. Other occurences of the Huber loss in RL include the training of ecological behavior in front of red traffic lights [223], vehicle control (left/right turn, acceleration/deceleration, [354]), and in (simulated) environments for a mountain car and a lunar lander [43].

In contrast to the minimax approach where the adversarial realization is considered to be the worst from a given uncertainty set, distributionally robust optimization optimizes an expectation which is not computed w.r.t. the ideal distribution but w.r.t. a set of distributions that contains the ideal distribution. In light of Sec. III, one can interpret this as an optimization of an expectation w.r.t. a contamination ball (although, in the literature, one not necessarily uses the classical contamination balls from Robust Statistics). In [132] (although actually not an RL approach, as there is no learned policy but the optimization problem is solved periodically with new data), the goal is to manage a fleet of electric vehicles according to the mobility demand and charging requirements. Here, the worst-case expected cost (w.r.t. sets of distributions that model the demand and the supply, respectively) is minimized over the number of dispatched vehicles across the regions. Here, the sets of distributions are confidence sets, estimated from historical data.

[282] consider state measurement errors and formulate the idea of smoothness regularizers that should encourage the differences between $\pi_\theta(\boldsymbol{s})$ and $\pi_\theta(\tilde{\boldsymbol{s}})$ to be small if the difference of $\boldsymbol{s}$ and $\tilde{\boldsymbol{s}}$ is small. Assuming $\boldsymbol{s}\in\mathbb{R}^p$, the smoothness regularizer has the form

$$R_s(\pi_\theta) = \mathbb{E}_{\boldsymbol{s}\sim P_S^{\pi_{\theta_t}}}[\max_{\tilde{\boldsymbol{s}}\in B(\boldsymbol{s},\epsilon)}(D(\pi_\theta(\boldsymbol{s}),\pi_\theta(\tilde{\boldsymbol{s}}))]$$

for the state visitation distribution P_S^π induced by a policy π an the l_p-ball $B(\boldsymbol{s},\epsilon)$ around $\boldsymbol{s}$ with radius ϵ. For the distance D, they use the Jeffrey's divergence

$$D_J(P||Q) = \frac{1}{2}D_{KL}(P||Q) + \frac{1}{2}D_{KL}(Q||P)$$

for stochastic policies and the Euclidean norm for deterministic policies. The agent is then trained w.r.t. the objective to maximize the Q-function, penalized by the smoothness regularizer.

[131] consider multi-agent RL with electric vehicles and argue that the individual charging patterns lead to additional model uncertainties with the goal to distribute the electric vehicles fairly among different regions while allocating low-battery vehicles to charging stations, minimizing the overall costs. For a cost function c, denote the worst-case state value function by

$$v_c^\pi(s) = \min_{\mathrm{J}\in U_c(\mathrm{J}_0,r)} \left(\mathbb{E}_\pi \left[\sum\nolimits_{t=1}^\infty \gamma^{t-1} c(s_t,a_t) \middle| s_1 = s \right] \right)$$

for the uncertainty set $U_c(\mathrm{J}_0,r) = \bigotimes_{s\in S, a\in A} \mathrm{J}(s,a)$ where each $\mathrm{J}(s,a)$ is a convex contamination ball around the true transition distribution $\mathrm{J}_0(s,a)$ with contamination radius r, i.e.,

$$\mathrm{J}(s,a) = \{(1-r)_+ \mathrm{J}_0(s,a) + \min(1,r)Q \,|\, Q\in\mathcal{M}_1(\mathcal{S}\times\mathcal{A})\}.$$

The objective for finding the optimal policy is then

$$\max_\pi (v_r^\pi(P_S)) \quad \text{s.t.} \quad v_c^\pi(s) \geq \tau \;\forall s\in S,$$

for some threshold τ and

$$v_c^\pi(s) = \min_{\mathrm{J}\in U_c(\mathrm{J}_0,r)} \left(\mathbb{E}_\pi \left[\sum\nolimits_{t=1}^\infty \gamma^{t-1} R_t \middle| s_1 = s \right] \right)$$

and $v_*^\pi(P_S) = \mathbb{E}_{s\sim P_S}[v_*^\pi(s)]$ for $*\in\{c,r\}$.

B. IMITATION LEARNING

Robust approaches for RL also carry over to IL.

Random perturbations of trajectories, which can be understood as an untargeted adversarial training, have been considered for IL in [19] who trained an autonomous driving agent based on expert trajectories. In [298], one perturbs the action selected by the agent by an adversarial action that should drag the vehicle from the intended path. In their setting, one is provided with future states by an expert. An inverse dynamics model (IDM) is applied to find suitable actions that allow the vehicle to attain these states. The policy is trained according to the reward function

$$-||\boldsymbol{a}_{\text{IDM}} - (\boldsymbol{a}_\theta + \boldsymbol{a}_\omega)||^2,$$

for the policy's action $\boldsymbol{a}_\theta$ and the adversarial action $\boldsymbol{a}_\omega$, so the agent should learn to imitate the optimal action by adjusting for the adversarial action. In order to decourage too harsh adversarial actions, they only inject an adversarial action with a certain probability and further restrict the adversarial action to a certain interval.

[183] consider a minimax criterion where an uncertainty set around the true observation is considered. This uncertainty set is given by coordinate-wise l_1-balls around the true state component. They consider simulated driving scenarios with traffic lights and intersections.

[138] introduce an IL algorithm based on GANs. They start with a regularized form of inverse RL with the objective

$$\text{argmax}_c(-J(c) + \min_\pi(-H(\pi) + \mathbb{E}_\pi[c(s,a)]) - \mathbb{E}_{\pi_E}[c(s,a)]),$$

where c denotes a cost function, H the γ-discounted causal entropy and π_E the expert policy. In order to prevent overfitting, they propose to regularize this objective with a convex

regularizer J, inducing the additional term $-J(c)$. Denoting this regularized objective by $\mathrm{IL}_J(\pi_E)$, and considering the RL objective

$$\mathrm{RL}(c) = \mathrm{argmin}_\pi(H(\pi) + \mathbb{E}_\pi[c(s,a)]),$$

[138, Prop. 3.2] shows that

$$\mathrm{RL} \circ \mathrm{IL}_J(\pi_E) = \mathrm{argmin}_\pi(-H(\pi) + J^*(\mathrm{OM}_\pi - \mathrm{OM}_{\pi_E})),$$

for the occupancy measure

$$\mathrm{OM}_\pi(s,a) = \pi(a|s)\sum_{t=0}^{\infty}\gamma^t P(s_t = s|\pi).$$

Now, [138] connect GANs with RL with the choice

$$J(c) = \begin{cases} \mathbb{E}_{\pi_E}[g(c(s,a))], & c < 0 \\ \infty, & c \geq 0 \end{cases},$$

$$g(x) = \begin{cases} -x - \ln(1-e^x), & x < 0 \\ \infty, & x \geq 0 \end{cases},$$

for which

$$J^*(\mathrm{OM}_\pi - \mathrm{OM}_{\pi_E}) = \max_D(\mathbb{E}_\pi[D(s,a)] + \mathbb{E}_{\pi_E}[\ln(1-D(s,a))]),$$

where D is taken from the set of all discriminative classifiers on $\mathcal{S} \times \mathcal{A}$. This leads to the task of finding a saddle point of

$$\mathbb{E}_\pi[\ln(D(s,a))] + \mathbb{E}_{\pi_E}[\ln(1-D(s,a))] - \lambda H(\pi).$$

With parametrizations D_ω and π_θ, this task can be solved in a GAN-style by alternatingly updating the parameters for the discriminator and the policy, respectively.

[220] propose to induce a Lipschitzness of both the discriminator and the policy by replacing the entropy regularizer $H(\pi)$ with the regularizer

$$R(D) = \frac{1}{|D|}\sum_{(s,a)\in D}|D_\omega(s + \delta_{s,a}a) - D_\omega(s,a)|,$$
$$\delta_{s,a} = \mathrm{argmax}_{||\delta||_2 \leq r}(|D_\omega(s+\delta,a) - D_\omega(s,a)|),$$

for training data D and discriminator network D_ω. The motivation is to better cope with observation noise. Their method is applied to robot locomotion.

[176] train an adversarially robust IL agent via a minimax game where the adversary aims at minimizing the objective the agent aims at maximizing (the entropy-regularized advantage function). In order to stabilize training, they suggest regularizing the objective of the ego-agent by a distillation loss term.

Similarly, [316] propose to alternatingly train an IL agent and an adversary where the latter learns to perturb the states in order to let the agent fail. Let the attack policy π_{adv} assign a density $\pi_{\mathrm{adv}}(\cdot|s)$ to a state s in order to produce some adversarial state s'. They then distinguish between sensory attacks, where the observed states are perturbed, or physical attacks, where the state itself is perturbed, resulting in a perturbation of the observed state and letting the transition model produce the next state based on the perturbed state. The objective is then to learn a policy under all possible attacks, i.e.,

$$\min_\pi(\max_{\pi_{\mathrm{adv}}}(J(\pi,\pi_{\mathrm{adv}})))$$

for

$$J(\pi,\pi_{\mathrm{adv}}) = \mathbb{E}\left[\sum_t \rho(a_t,\pi_E(s_t))\middle| a_t \sim \pi(\pi_{\mathrm{adv}}(s_t)), s_{t+1} \sim \mathcal{I}(\cdot|s_t,a_t)\right]$$

in the case of sensory attacks, or

$$J(\pi,\pi_{\mathrm{adv}}) = \mathbb{E}\left[\sum_t \rho(a_t,\pi_E(s_t))\middle| a_t \sim \pi(\pi_{\mathrm{adv}}(s_t)), s_{t+1} \sim \mathcal{I}(\cdot|\pi_{\mathrm{adv}}(s_t),a_t)\right]$$

in the case of physical attacks, respectively, for some loss function ρ that penalizes the discrepancy between two actions.

A typical approach from Robust Statistics, namely a robustification of the objective, is done in [240] who consider value function estimation and who point out that this is usually done by minimizing the squared Bellman error, i.e., for Bellman operator

$$(\mathcal{T}V)(s) = \mathbb{E}[R_{t+1} + \gamma^{t+1}V(s_{t+1})|s_t = s],$$

this objective is given by

$$\min_\theta\left(\sum_{s\in\mathcal{S}} P_S(s)((\mathcal{T}V_\theta)(s) - V_\theta(s))^2\right)$$

for some distribution P_S on $\mathcal{S}$ and an approximation V_θ of the true function V^π for which $(\mathcal{T}V^\pi) = V^\pi$ holds. [240] propose to replace the squared loss by the absolute or the Huber loss and show how these new objectives can be minimized.

[198] consider IL if contamination in the classical Tukey-Huber sense are allowed, i.e., in the pool of offline demonstration data, a fraction of ϵ of the instances (state-action pairs) can be arbitrarily corrupted (note that this fraction is deterministic as in the BDP context, not stochastic as in convex contamination settings). They propose to randomly partition the data into B batches and to use the mean of the likelihoods in each batch as the objective, so the overall objective is the median of the means for which they propose a gradient-based optimization algorithm. This contamination model has also be considered in [365], [366] for RL with policy gradient. More precisely, in [366], an ϵ-fraction of transitions can be modified arbitrarily while in [365], both an ϵ-fraction of rewards and transitions can be perturbed arbitrarily, therefore, they call this contamination scheme "strong data corruption".

[366] point out that many of the existing robust RL methods consider offline RL, where they distinguish between online RL, i.e., an adversary can adapt their perturbation in each iteration, and offline RL, where the contamination must be generated prior to training. In [365], in contrast, the ϵ-contamination scheme is designed for online learning in the sense that the adversary can decide in each iteration whether to replace the current reward and the new state with arbitrary values, with the restriction that this can only be done in at most $\epsilon m_{\mathrm{iter}}$ training iterations if m_{iter} is the maximum number of iterations.

[70] assume that reference trajectories or parts of reference trajectories are adversarial in the sense that they accomplish the task with illegal means. Having a small initial set of guaranteed benign trajectories, they detect adversarial trajectories by a divergence measure. To this end, they partition trajectories during training into parts corresponding to sub-tasks and learn sub-policies (options, see [295], [18]) for each sub-task. In order to be able to detect adversarial trajectories that only differ from benign ones in some time steps, they propose to use the occupancy measure w.r.t. the clean trajectories, i.e.,

$$\mathrm{OM}_{\pi}^{\mathrm{clean}}(s,a) = \sum_{(s_i,a_i)\in\tau_{\mathrm{clean}}} \pi^*(a_i|s_i) \sum_t \gamma^t P(s_t = s_i|\pi_{\mathrm{clean}})$$

for a clean trajectory τ_{clean} generated by a clean policy π_{clean} and for the optimal policy π^*. Because this measure is zero if two trajectories are very close without overlapping, [70] combine it with the Fréchet distance

$$\mathrm{FD}(\tau) = \min_{\alpha,\beta}(\max_{t\in[0,1]}(||\tau(\alpha(t)) - \tau_{\mathrm{clean}}(\beta(t))||_2))$$

for functions $\alpha,: [0,1] \to \{0,1,...,|\tau|\}$, $\beta,: [0,1] \to \{0,1,...,|\tau_{\mathrm{clean}}|\}$. Then, a classifier is trained on the two scores in order to decide whether the trajectory part is adversarial or benign.

C. MODEL-PREDICTIVE CONTROL

In MPC, one assumes a model for the system dynamics, i.e.,

$$s_{t+1} = f(s_t, u_t, \nu_t),$$

where u_t are the control inputs to the system, which are contained in some space U, and control noise ν_t. In its simplest form, an MPC problem is given by

$$\min_{\boldsymbol{u}\in U}\left(\sum_t \rho(r_t)\right) \quad \text{s.t.} \quad s_{t+1} = f(s_t, u_t),$$

for the control errors $r_t = d(s_t^{\mathrm{sp}}, s_t(u_t))$ where s_t^{sp} is the set-point that the agent should follow at time t and where d is some distance measure. The objective is potentially conditioned on other constraints such as that the states and control inputs should, at least with a certain probability, be contained in some subspace of $\mathcal{S}$ and U, respectively, that correspond to safe or comfortable behavior, or penalized by a term that discourages the control inputs to vary too much over time.

Similar approaches as for RL and IL have also been introduced for MPC, e.g., replacing the quadratic loss for the control errors by the absolute loss as in [79] and [227], by the Cauchy loss [77] or the dynamically scaled covariance loss introduced in [4], which has been used in [77] as loss function for MPC.

A minimax formulation of an MPC in order to deal with worst-case uncertainties goes back to [41]. [362] propose another minimax formulization where the objective contains worst-case losses w.r.t. an uncertainty set on the (discrete) behavior of surrounding vehicles, given as a probability simplex. Experiments on simulated data with a time step of 0.1s and a planning horizon of 5s confirm real-time capability. [226] consider the control of maritime vessels and aim at avoiding collisions. Here, the obstacles are overapproximated by balls, and due to tidal effects, their radius changes, i.e., they are modelled as random variables. Assuming that an empirical distribution for each radius exists, their robust MPC approach considers an uncertainty set in the form of contamination balls around each empirical distribution, based on the p-th order Wasserstein distance.

[228] point out that robust control approaches often include robust optimization where the constraints are only known up to some noise term with the goal to keep the constraints satisfied for all possible noise terms for a given uncertainty set, i.e., $g(x,\xi) \leq 0$ for all $\xi \in \mathcal{U}$ for some uncertainty set $\mathcal{U}$ and some function g, making the approach rather conservative. The other approach is distributionally robust optimization where the supremal expectation of the constraints have to be satisfied w.r.t. a given set of distributions w.r.t. which the expectation is computed, i.e.,

$$\sup_{P\in\beth}(\mathbb{E}_P[g(x,\xi)]) \leq 0$$

for some set $\beth$ of distributions. Motivated by the functional view from Robust Statistics, [228] consider the supremum bias due to the decomposition

$$\sup_{P\in\beth}(\mathbb{E}_P[g(x,\xi)]) = \mathbb{E}_{\hat{P}}[g(x,\xi)] + \sup_{P\in\beth}(\mathrm{Bias}(P,g,\hat{P}))$$

with

$$\mathrm{Bias}(P,g,\hat{P}) = \int g d(P-\hat{P}).$$

They compute the supremum bias for the commonly used 1-th order Wasserstein and MMD metric, resulting in

$$\sup_{W_1(P,\hat{P})\leq\epsilon}(\mathrm{Bias}(P,g,\hat{P})) = \epsilon L_g$$

for Lipschitz constant L_g of g w.r.t. the first argument, and

$$\sup_{\mathrm{MMD}(\mathcal{H}_g,P,\hat{P})\leq\epsilon}(\mathrm{Bias}(P,g,\hat{P})) = \epsilon||g||_{\mathcal{H}_g}$$

for the RKHS $\mathcal{H}_g$ of g, respectively. Due to L_g and $||g||_{\mathcal{H}_g}$ being unknown in practice, they propose to control the distributional robustness for the 1-th order Wasserstein distance by

$$\epsilon \max_i(||\nabla_x g(x_i,\cdot)||),$$

motivated by $L_g \geq \sup_x(||\nabla_x g(x,\cdot)||)$, and for the MMD distance, they prove that the original inequality w.r.t. the supremal expectation is satisfied if there exists $h \in \mathcal{H}$ for some RKHS $\mathcal{H}$ such that

$$\max_i(h(\xi_i)) + \epsilon||h||_{\mathcal{H}} \leq 0 \text{ and } g(x,\xi) \leq h(\xi)\ \forall\xi\in\mathcal{U}.$$

Another common robust MPC strategy is tube-based MPC where one assumes that there is a function $g : 2^{\mathcal{S}} \times U \times 2^{\mathbb{R}^q}$ such that $g(\boldsymbol{s}_t, \boldsymbol{u}_t, \boldsymbol{\nu}_t) \in g(S, \boldsymbol{u}_t, N)$, where $\boldsymbol{\nu}_t \in \mathbb{R}^q$. Furthermore, if $\mathcal{S}_1 \subset \mathcal{S}_2$, it holds that $g(\mathcal{S}_1, \boldsymbol{u}, \mathbb{R}^q) \subset g(\mathcal{S}_2, \boldsymbol{u}, \mathbb{R}^q)$ for all $\boldsymbol{u} \in U$ (e.g., [287]). In other words, the set of all possible forward reachable states is over-approximated by a tube.

[287] enhance tube-based robust MPC with collision avoidance constraints and train an autonomous agent for a car. [26] apply robust tube-based MPC to the training of an autonomous driving agent in order to avoid collisions with pedestrians, both uncertain static and uncertain dynamic pedestrians. [25] consider MPC that satisfies safety constraints such as collision avoidance or terminal conditions like a full stop or parking, and allow for the state and controller inputs being as close as possible to a reference input. In simulations with a time step size of 0.05s and a prediction horizon of 20s, they confirm real-time performance of their controller. [230] consider a robust tube-based MPC for lane keeping of an autonomous vehicle, [142], [215], [157], [179], [350] (experiencing quicker convergence than standard MPC in their simulations), [20], [57] apply it to path tracking, and [340] to overtaking. [98] consider obstacle avoidance using tube-based MPC on icy and snowy roads and confirm real-time capability in their experiments. [290] use tube-based MPC for collision avoidance with moving obstacles. [222] use tube-based MPC in order to let multiple agents satisfy platooning requirements, i.e., maintaining the same speed. [244] combine l_1-adaptive control, which lets the system behave as a linear model, disregarding uncertainties and perturbations, with tube-based MPC. Delay (even time-varying) as an additional source of uncertainty is also considered, for example in [199], who use tube-based MPC apply it to steering, and [163], who consider uncertainties in timing due to multiple sources simultaneously, formulate the problem as tube-based MPC, and perform experiments concerning static collision avoidance and overtaking. [329] consider tube-based robust MPC for autonomous racing.

H_∞-control considers the H_∞-norm of the transfer function G of a linear state space model, i.e.,

$$||G||_\infty = \sup_\omega(\sigma_{\max}(G(j\omega)))$$

for the maximum singular value $\sigma_{\max}$. Optimal H_∞-control considers minimizing $||T_{zw}||_\infty$ where T_{zw} denotes the upper left block of the transfer matrix G, however, as the solution is often not unique and difficult to compute, one relaxes the problem often to satisfying $||T_{zw}(s)||_\infty < \gamma$ for some $\gamma > 0$. [275] use H_∞-control for adaptive cruise control and lane change in queues, [116] path tracking, cruise control and lane change of electric vehicles, [159] consider double lane change and serpentine maneuvers for electric vehicles, [342], [253] steering while driving at constant speed, [160] path following and lateral stability of autonomous electric vehicles, [165] path following and lateral stability of autonomous vehicles, [236] speed and current control for electric vehicles, [242] collision avoidance, [261] lane-keeping, [95] lateral control, and [137], [377] path-tracking.

[278] consider lane change on highways and model the lane switching behavior of the surrounding vehicles as Markov jump process. Let ω be a set of parameters that model the uncertainty of the system. Then, for a sample $\{\hat{\omega}_i\}$, they compute the empirical distribution $\hat{P}_n$ and let the ambiguity set be the contamination ball around $\hat{P}_n$ w.r.t. the TV distance. The robust MPC approach then solves a minimax problem, i.e., finding the control sequence that minimizes that maximum cost over all such distributions.

See also [355] for further references on robust MPC approaches for autonomous driving.

D. BYZANTINE ROBUSTNESS

Federated and distributed (reinforcement) learning is done for a lot of recent autonomous driving models, see [319] and references therein. Therefore, it is important to ensure Byzantine-robustness of those approaches. [319] themselves craft poisoning attacks against federated learning in a non-linear, autonomous steering control scenario, [102] propose attacks against trajectory prediction via federated learning.

As for distributed RL, e.g., [85] consider Byzantine-robustness and suggest classical outlier detection in each learning round by computing the mean of the medians of the estimated policy gradients of each agent and neglecting those which differ by a least two standard deviations from this estimate. [80] consider a general bandit algorithm and allow for a constant ϵ-fraction of agents to be byzantine, and propose to compute the shortest interval containing a fixed fraction of rewards so that the mean reward is then given by the mean of the rewards contained in this interval. [58] consider also a trimmed mean in order to estimate the value function in online and offline distributed RL. [208] propose a poisoning scheme for federated RL and assume that the attacker can perturb the observations of some of the trained RL agents, but has no information about the underlying MPC. They also consider corrupted critic networks in actor-critic RL. Note that in these settings, the individual distributed agents themselves take the role of the instances in the original understanding of case-wise contamination.

Further approaches for Byzantine-robustness that are not directly tailored to federated RL are given in the literature. For example, [35] show that no linear aggregation is Byzantine-robust if one single local model is poisoned. Note that analogy to the non-robustness of the mean or the non-robustness of Bagging. They propose Krum, a technique where essentially a variant of the geometric median is computed, more precisely, the local model with the smallest distance to its nearest neighbors. They show that Krum guarantees Byzantine robustness if the fraction of malicious models is smaller than $(n/2 - 1)$, so one could interpret the fraction of 0.5 of malicious models as "Byzantine-BDP" here. The notion of such a BDP has

recently been introduced in [113]. Variants of the geometric median have also been considered for example in [186], [321], [286], or in [56] and [271] where a geometric median of means of gradients is proposed. [351] consider median and trimmed-mean aggregation of the coordinate-wise gradients reported back from the local learners and analyze statistical error rates. [379] improve their work as the bounds in [351] depend on the dimensionality so that the rates may be sub-optimal in high dimensions and also interpret the minimum fraction of Byzantine models that lead to unreliability of the federated learning procedure as BDP.

See [309] for a recent overview of robust federated learning.

VI. OUTLOOK AND FUTURE WORK

In this section, we outline some ideas of robust strategies that were not yet fully applied to autonomous driving tasks. Moreover, we provide suggestions for benchmarking studies, where different robust algorithms for each of the individual application areas could be compared in order to assess whether some classes of robust algorithms are better suited than other, and to get an intuition about the amount of contamination in typical data sets from the respective area.

A. FURTHER STRATEGIES FROM ROBUST STATISTICS

The concept of influence functions is rarely seen in the context of autonomous driving in the sense of Robust Statistics, however, the term "influence function" is often used in a physical sense, i.e., quantifying the impact of one physical variable to some property. Influence functions are used as diagnostic tools in deep learning in general, for example, in [174], and they are one of many approaches of XAI (e.g., [16]). [106] propose adversarial attack against influence functions and show that the interpretation of a NN based on the influence function is also fragile and highly vulnerable to adversarial attacks.

A particular application in autonomous driving is given in [301] where influence functions are used in order to predict the impact of a data point on pedestrian detection. More precisely, the influence function is used as proxy in order to predict the differences between the test losses for a model trained on the original data and a model trained on data where one instance has been deleted.

Apart from the quantification of the impact of an observation on the estimator, diagnostics based on influence curves can be used to generally strengthen the understanding of the data, for example, whether there are clusters of points with high impact or by trying to find particular properties that make data points influential. This strategy may not only be applied for perception but also for planning in the sense that certain actions or whole trajectories of adversarial agents are identified as influential on the RL training result.

Nevertheless, apart from diagnostic purposes, the influence curve can also be used in order to robustify an estimator directly through the perspective of local robustness. The "robust losses" introduced in Sec. III-D themselves induce bounded influence curves of the corresponding M-estimator by Eq. 2, however, it is also possible to directly robustify the influence curves. To this end, a so-called asymptotic linear expansion of the estimator in the form

$$\hat{\boldsymbol{\theta}}_n = \hat{\boldsymbol{\theta}}_0 + \frac{1}{n}\sum_{i=1}^{n} \mathrm{IC}(\boldsymbol{X}_i, T, P) + \mathbf{rem}$$

must be valid, for some consistent initial estimator $\hat{\boldsymbol{\theta}}_0$ and a remainder term **rem** (see, e.g., [259]). This property holds, for example, for asymptotically normal M-estimators [259], SVMs [120], or regularized M-estimators [324]. Given an influence curve, one can formulate different optimization problems in order to robustify the underlying estimator, for example, minimizing the covariance of the influence curve subject to a bound on the bias, minimizing the MSE [259], or finding the estimator the achieves maximum asymptotic relative efficiency even under the worst-case contamination radius [260]. Such "optimally-robust" estimators do not seem to have been considered so far for applications in autonomous driving, but would potentially increase the performance of the trained models compared to those trained according to the classical robust losses.

Another topic, which becomes increasingly important when dealing with high-dimensional data, is variable selection. Robust variable selection algorithms already have been proposed in the literature, for example, the sparse LTS [7], robust Boosting variants [206], [161], or a trimmed Stability Selection [326]. Such techniques could be applied in situations with high-dimensional state spaces in order to identify relevant variables.

In the reviewed literature, except for those considering Kalman filters where contamination in the innovations has already been regarded, one does not distinguish between contamination in the responses and contamination in the regressor variables. This distinction is important because one can argue that outliers in the responses, given clean predictor columns, may naturally be bounded or that large errors may be detectable in advance, implying that unbounded loss functions with a bounded gradient such as the Huber loss function are not problematic here. However, outliers in the predictor columns are known to be more challenging. This situation would even be natural, as the predictor variables are also measurements in some applications such as location estimation via GPS data, where coordinates or clock offsets enter as predictor variables, or vehicle parameter estimation such as tractive forces or electric parameters, where variables such as velocity or voltages are used as predictor variables. Moreover, contamination in the predictor variables occurs when one cannot assume perfect reference data. For example, bounding box estimation is often accompanied with ground truth coordinates, which may be imperfect due to errors done by human annotators, or the reference trajectories in imitation learning are corrupted. Here, robust estimation techniques that allow for this type of contamination, which is more chal-

lenging than just considering contamination in the responses, allow for addressing such situations.

Large measurement errors that are a source of contamination in the data are especially problematic in data with a high number of variables and a rather low number of observations. Here, the fraction of contaminated instances can quickly become very large, as a single contaminated cell already makes an instance an outlier (e.g., [10]). Therefore, cell-wise robust approaches have recently been proposed in the literature, i.e., algorithms that can cope with a certain fraction of contaminated cells, even if all instances would be contaminated. For example, there are cell-wise robust counterparts of location and scatter matrix estimation algorithms [5], [180], regression [39], [181], [90], and clustering [100]. In high-dimensional data with an admissible cell-wise contamination scheme (which could be a random selection of perturbed cells as, e.g., having contamination on the response column only would not allow for any advantage of cell-wise robust over case-wise robust algorithms, see [326]), the application of cell-wise robust procedures could improve the robustness against large case-wise contamination rates, even potentially becoming an alternative to RANSAC and its variants.

Outlier detection algorithms have been applied in advance to the data in many references. Unless one faces univariate samples, one should be aware of several pitfalls when trying to detect outliers in multivariate data (see, e.g., [121]). A single large outlier in a multivariate sample can make other outliers invisible, essentially due to deforming the confidence region in a way that other outliers fall within this region, which is referred to as "masking effect". Similarly, non-contaminated observations can, due to the same reason, be located outside the confidence region so that they are incorrectly flagged as outliers, which is referred to as "swamping effect". Therefore, one should not apply a classical outlier detection algorithm on the data set once and consider the non-flagged observations as clean. In addition, when assuming an underlying model, as in regression, simply detecting outliers in a model-agnostic way is unlikely to find instances whose entries are insuspicious but which appear as outliers w.r.t. the assumed model. In such settings, model-based outlier detection, which is essentially done in the iterations of LTS, is necessary. As for cell-wise contamination, some strategies for detecting and imputing cell-wise outliers have been proposed, e.g., [263], [255].

B. SUGGESTIONS FOR BENCHMARK STUDIES AND FUTURE RESEARCH

This survey paper has provided an overview of robust methods in autonomous driving in a comprehensive way and in a unified notation. It should not only serve as a detailed overview for researchers and practitioners, but also pave the way for organizing and conducting benchmarking studies, which are of interest on their own, but beyond the scope of this work.

For nearly each application area, there are already several robust strategies in the literature. Of course, in the respective papers, comparisons already have been made, but often with some selected comptetitors from the literature. The comparison of a large number of algorithms for one specific problem would be desirable. In particular, the navigation section contains a plethora of robust algorithms that follow different strategies: robust criteria, noise modeling or clipping. While robust criteria and clipping follow a similar idea as robust regression approaches, namely downweighting or even neglecting outlying instances during optimization, noise modeling learns the noise distribution and integrates it into the computation of the posterior distribution for the next state. Both strategies are accompanied with advantages and disadvantages. Noise modeling, in particular when using online algorithms such as variational inference, allow for non-stationary noise distributions and have already been successfully applied in real-world settings where both the measurement and the process noise were heavy-tailed, i.e., both additive and innovation outliers occurred. The computation of the posterior distribution allows for the quantification of estimation uncertainty. On the other hand, noise modeling requires assumptions about the noise distributions, while robust optimization criteria are usually applicable under milder assumptions. Therefore, algorithms from each type should be compared on data with both additive and innovation outliers and where the noise distributions are non-stationary. The iterative nature of both variational inference and the optimization of robust criteria usually induces computational overheads compared to a non-robust algorithm. It should be investigated how this overhead scales on real-world data and in dependence of the amount of contamination.

Benchmarking studies could be both based on simulations and real data. As for simulations, the advantage is that via data generation, one can directly control the type and the amount of contamination, for example, case-wise contamination with outliers only in the responses, only in the regressors, or both, or cell-wise contamination. This would enable to validate different robust algorithms concerning breakdown. Although one can compute theoretical breakdown points, one should be aware of the facts that on the one hand, the breakdown point corresponds to a worst-case scenario, indicating that an estimator does not necessarily break down once the corresponding fraction of instances or cells is contaminated. On the other hand, the algorithm with which the estimator is computed usually is not regarded when computing a breakdown point, therefore, due to numerical pitfalls such as vanishing gradients, it is possible to have an earlier breakdown than expected. Such an effect has been observed in [327] for neural network training.

As for real data, any type of data base with real data and a benchmarking study on such data is of course also of interest on its own. A future benchmarking study should focus on the comparison of the performance of robust algorithms from each individual subsection in this review paper, including non-robust competitors. In particular, on real data, one cannot determine the true underlying contamination model nor the contamination radius. Therefore, one usually applies both a

robust and a non-robust method and decides to use the non-robust estimator henceforth if their performance does not differ much or if the classical estimator performs better. In a more granular approach, for any algorithm whose robustness can be controlled, e.g., by the trimming parameter that decides on the size of the clean subset used for fitting, one should apply the respective algorithm with different robustness properties in order to get a better understanding of the amount of contamination of the data in the real world.

The same argumentation holds for adversarial and Byzantine robustness. In particular, for Byzantine robustness, each contaminated gradient/model/input from an adversarial machine takes the role of one contaminated instance in a standard data set. As for adversarial robustness, from the perspective of Robust Statistics, one could ask whether a worst-case analysis is possible, and how the amount of contamination could be quantified. Both questions should be addressed in future research.

If a worst-case perspective, without any restrictions, would be pursued, one would likely consider adversarial actions from an adversarial agent where the worst case would correspond to the maximum acceleration or jaw. However, in particular for planning algorithms, there is one crucial aspect when it comes to perturbations/adversarial actions: Realism. This property has already been identified as important in the literature on adversarial attacks on images, e.g., [308]. It is well-known that training strategies such as RL fail if the environment is too challenging, e.g., [298]. Although such edge cases are important for the safety assessment of an autonomous driving system, focusing on them appears not to be the correct way to assess the robustness of the learning algorithm itself in the context of breakdown. Moreover, one could ask how the amount of contamination should be quantified here, i.e., whether one should count each adversarial action or each adversarial trajectory.

In order to define a breakdown for learned policies, one should check whether the existing definitions of a breakdown from Robust Statistics can be translated. In Robust Statistics, when learning a parametrized model, a breakdown indicates that the learned parameters can be arbitrarily close to the borders of the underlying parameter set, or made arbitrarily large. It should be assessed whether this is possible for the parameters learned when training a parametrized policy, or the control inputs learned in MPC, as well as the implications on the behavior of the agent. Moreover, robustness should not be confused with safety here. For example, consider an autonomous car and a distribution that is learned on its action space. If the environment is extremely challenging, it can learn just not to move. From the perspective of safety, such a behavior could be interpreted as excellent, but from the perspective of Robust Statistics, it could be interpreted as breakdown since it would correspond to (or be close to, depending on the algorithm) a Dirac distribution on a particular value (zero acceleration). Apart from the quantitative robustness that corresponds to the influence curve and the breakdown point, [122] propose a notion of qualitative robustness of an estimator, which indicates that small changes in the underlying distribution (usually measured in the Prokhorov distance) imply only small changes in the estimates. Regularity of the trained policy has already been considered in [220], [282], where one focuses on the regularity w.r.t. state perturbations. A unified approach would consider a joint distribution on the observed states (e.g., due to measurement errors or in partially observable settings) and the transitions. An approach in order to assess qualitative robustness, in a simulated setting, could be to identify realistic but challenging scenarios, such as complex urban environments or severe weather conditions, and to gradually change the underlying distribution to increase the mass of such challenging situations, or, in other words, to contaminate an "ideal distribution", which should reflect the real world as much as possible, resulting in a contaminated distribution, similarly as in [131], who concentrated on the transition distribution. One may be able to identify when the trained policy starts to deteriorate in the sense that a certain regularity property of the underlying policy no longer holds once the distribution favors challenging situations too strongly. This amount of contamination could then be interpreted as the BDP.

In the reviewed literature, a robust method for either a particular perception or planning area has been presented. A complete autonomous system however is composed by several modules, at least multiple perception modules and a planning module. It is vital to consider the robustness of the whole system. Usually, one may argue that the whole system is not robust if at least one module is not robust. On the other hand, applying robust methods for each subsystem may have the disadvantage of a low overall efficiency, as a consequence from the robustification. For example, [279] consider a state-space model in order to incorporate observation noise. This observation noise itself depends on the perception module. Therefore, one should examine to what extend a certain non-robustness in the perception modules could be compensated during the training of a driving policy.

Once an algorithm has proven to be competitive in terms of its generalization performance, accuracy, or security, a crucial question is whether the running time is sufficiently low in order to be applicable in real-world autonomous driving scenarios. In particular, many of the individual tasks require real-time performance. While robust algorithms tolerate contamination which may let non-robust algorithms break down, they are less efficient than non-robust algorithms, so that more training data are required for convergence. The major drawback however is that the optimization of a non-convex loss function, a minimax loss, or, for adversarial training, the optimization of an adversary, may be required, usually resulting in a considerably higher computation time, which could be the main hindrance for real-world applications, regardless of their performance in terms of safety or security. Several of the reviewed papers already confirm in their experiments, some even on real-world data, that their robust algorithm has real-time performance. Nevertheless, research on the real-time

capabilities of robust algorithms is necessary, in particular when considering the whole autonomous system with many individual modules.

As for the whole system itself, an even broader research question is the interaction between different properties that a system should satisfy, in particular, safety and security. As for security, the reviewed robust approaches for planning as well as for Byzantine robustness already allow for training with adversarial attacks, directly corresponding to security. For perception where training data are given, adversarial robustness does not fall within the scope of Robust Statistics as adversarial attacks are crafted after model training, while Robust Statistics considers the training procedure itself. As explained in Sec. III, poisoning attacks in contrast are crafted before model training. However, as poisoning attacks are usually restricted by a geometric bound, methods from Robust Statistics can be assumed to fail if the majority of the instances is perturbed. Nevertheless, a potential topic for future research could be the interaction between adversarial robustness (including poisoning attacks) and robustness in the sense of Robust Statistics, i.e., whether methods from Robust Statistics already increase adversarial robustness or how models that are both robust in the sense of Robust Statistics and adversarially robust can be trained. A combination with Byzantine robustness, i.e., training a federated RL model with both Byzantine and adversarial attacks, could also be considered. As for safety in autonomous driving, the goal is to assess whether an autonomous system can perform undesired actions or with which probability such behavior occurs. Although, in particular for deep-learning-based systems, reasoning whether an outlier in the training data of some perception module caused an undesired action may not be possible, one could at least train robust and non-robust subsystems and evaluate whether the robustness of these subsystems affects the safety of the combined system.

While the overall system in the previous paragraphs corresponds to a single autonomous agent, another topic of interest is collborative navigation. If the individual agents communicate, there are multiple state measurements, depending on whether an individual is inside the lane of sight or at least sufficiently close to another agent. As each agent has a different location, it can happen that the perception for some agents suffers from anomalies such as lightning variations. Therefore, regarding the joint information, one has a similar setting as in federated RL. One may consider the corrupted measurements of one agent as Byzantine information. Of course, computing an average or distance-weighted average of the information would lead to corrupted joint information. Here, one may consider integrating a robust aggregation procedure, such as a trimmed mean, into the algorithm that infers the next states. Nevertheless, in this context, other problems additionally need to be solved, such as communication delays or data loss, as argued in [276], or correlated observations among the individual agents (e.g., [48], [205]). On top of that, [299] consider adversarial interference in the inter-agent communication.

ACKNOWLEDGMENT

The author would like to thank Peter Ruckdeschel, Martin Fränzle and Eike Möhlmann for helpful comments and/or discussions, as well as three anonymous referees for valuable comments that helped to improve the quality of the paper. Of course, the author is solely responsible for any errors.

REFERENCES

[1] G. Agamennoni, J. I. Nieto, and E. M. Nebot. An outlier-robust Kalman filter. In *2011 IEEE International Conference on Robotics and Automation*, pages 1551–1558. IEEE, 2011.

[2] G. Agamennoni, J. I. Nieto, and E. M. Nebot. Approximate inference in state-space models with heavy-tailed noise. *IEEE Transactions on Signal Processing*, 60(10):5024–5037, 2012.

[3] P. Agarwal. *Robust graph-based localization and mapping*. PhD thesis, Dissertation, Albert-Ludwigs-Universität Freiburg, 2015, 2015.

[4] P. Agarwal, G. D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard. Robust map optimization using dynamic covariance scaling. In *2013 IEEE International Conference on Robotics and Automation*, pages 62–69. IEEE, 2013.

[5] C. Agostinelli, A. Leung, V. J. Yohai, and R. H. Zamar. Robust estimation of multivariate location and scatter in the presence of cellwise and casewise contamination. *Test*, 24(3):441–461, 2015.

[6] M. A. Akram, P. Liu, Y. Wang, and J. Qian. GNSS positioning accuracy enhancement based on robust statistical MM estimation theory for ground vehicles in challenging environments. *Applied sciences*, 8(6):876, 2018.

[7] A. Alfons, C. Croux, and S. Gelper. Sparse least trimmed squares regression for analyzing high-dimensional large data sets. *The Annals of Applied Statistics*, 7(1):226–248, 2013.

[8] P. Almási, R. Moni, and B. Gyires-Tóth. Robust reinforcement learning-based autonomous driving agent for simulation and real world. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020.

[9] M. T. Alonso, C. Ferigato, D. Ibanez Segura, D. Perrotta, A. Rovira-Garcia, and E. Sordini. Analysis of 'pre-fit' datasets of gLAB by robust statistical techniques. *Stats*, 4(2):400–418, 2021.

[10] F. Alqallaf, S. Van Aelst, V. J. Yohai, and R. H. Zamar. Propagation of outliers in multivariate data. *The Annals of Statistics*, 37(1):311–331, 2009.

[11] A. Amini, I. Gilitschenski, J. Phillips, J. Moseyko, R. Banerjee, S. Karaman, and D. Rus. Learning robust control policies for end-to-end autonomous driving from data-driven simulation. *IEEE Robotics and Automation Letters*, 5(2):1143–1150, 2020.

[12] A. Angrisano and S. Gaglione. Mitigation of leverage observation effects in GNSS robust positioning. In *2018 IEEE International Workshop on Metrology for the Sea; Learning to Measure Sea Health Parameters (MetroSea)*, pages 278–282. IEEE, 2018.

[13] M. Aqel, M. Marhaban, M. Saripan, and N. Ismail. Review of visual odometry: types, approaches, challenges, and applications. *SpringerPlus*, 5(1): 1897–1922, 2016.

[14] I. Arasaratnam and S. Haykin. Cubature kalman filters. *IEEE Transactions on automatic control*, 54(6):1254–1269, 2009.

[15] M. U. I. Arif, M. Jameel, and L. Schmidt-Thieme. Directly optimizing IoU for bounding box localization. In *Asian Conference on Pattern Recognition*, pages 544–556. Springer, 2019.

[16] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al. Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115, 2020.

[17] B. W. Babu, S. Kim, Z. Yan, and L. Ren. σ-DVO: Sensor noise model meets dense visual odometry. In *2016 IEEE international symposium on mixed and augmented reality (ISMAR)*, pages 18–26. IEEE, 2016.

[18] P.-L. Bacon, J. Harb, and D. Precup. The option-critic architecture. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[19] M. Bansal, A. Krizhevsky, and A. Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. *arXiv preprint arXiv:1812.03079*, 2018.

[20] H. Bao and R. Huang. Tube-based model predictive control for unmanned vehicles trajectory tracking. In *2025 2nd International Conference on Electrical Technology and Automation Engineering (ETAE)*, pages 555–560. IEEE, 2025.

[21] J. Bao, X. Mu, X. Yu, Z. Zhu, and H. Qin. Outlier-robust underwater navigation using a dual-robust-kernel kalman filter. *IEEE Signal Processing Letters*, 2025.

[22] J. Bao, X. Yu, X. Mu, C. Hu, and H. Qin. Geometric extended kalman filter with dual robust kernels for integrated navigation. *IEEE Transactions on Instrumentation and Measurement*, 2025.

[23] J. T. Barron. A general and adaptive robust loss function. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4331–4339, 2019.

[24] O. Bastani, Y. Ioannou, L. Lampropoulos, D. Vytiniotis, A. Nori, and A. Criminisi. Measuring neural net robustness with constraints. In *Advances in neural information processing systems*, pages 2613–2621, 2016.

[25] I. Batkovic, A. Gupta, M. Zanon, and P. Falcone. Experimental validation of safe MPC for autonomous driving in uncertain environments. *IEEE Transactions on Control Systems Technology*, 2023.

[26] I. Batkovic, U. Rosolia, M. Zanon, and P. Falcone. A robust scenario MPC approach for uncertain multi-modal obstacles. *IEEE Control Systems Letters*, 5(3):947–952, 2020.

[27] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.

[28] A. Bellés and D. Medina. Robust PPP-AR using M-estimators for multi-fault scenarios. In *2025 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pages 251–257. IEEE, 2025.

[29] A. Bellés, D. Medina, P. Chauchat, S. Labsir, and J. Vilà-Valls. Robust error-state kalman-type filters for attitude estimation. *EURASIP Journal on Advances in Signal Processing*, 2024(1):75, 2024.

[30] A. Bellés, D. Medina, P. Chauchat, and J. Vilà-Valls. Reliable GNSS joint position and attitude estimation in harsh environments through robust statistics. In *2022 IEEE Aerospace Conference (AERO)*, pages 1–9. IEEE, 2022.

[31] P. Bergström and O. Edlund. Robust registration of point sets using iteratively reweighted least squares. *Computational optimization and applications*, 58:543–561, 2014.

[32] S. Beyer. *Robuste Parameterschätzung für Elektrofahrzeuge*. PhD thesis, Universitätsbibliothek der Universität der Bundeswehr München, 2019.

[33] M. J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *1993 (4th) International Conference on Computer Vision*, pages 231–236. IEEE, 1993.

[34] M. J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer vision and image understanding*, 63(1):75–104, 1996.

[35] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30, 2017.

[36] G. Blewitt, C. Kreemer, W. C. Hammond, and J. Gazeaux. MIDAS robust trend estimator for accurate GPS station velocities without step detection. *Journal of Geophysical Research: Solid Earth*, 121(3):2054–2068, 2016.

[37] X. Bo, A. A. Razzaqi, and L. Yalong. Cooperative localisation of AUVs based on Huber-based robust algorithm and adaptive noise estimation. *The Journal of Navigation*, 72(4):875–893, 2019.

[38] A. Borkar, M. Hayes, and M. T. Smith. Robust lane detection and tracking with RANSAC and kalman filter. In *2009 16th IEEE International Conference on Image Processing (ICIP)*, pages 3261–3264. IEEE, 2009.

[39] L. Bottmer, C. Croux, and I. Wilms. Sparse regression for large data sets with outliers. *European Journal of Operational Research*, 297(2):782–794, 2022.

[40] H. Calatrava, D. Medina, and P. Closas. Towards robust collaborative DGNSS in the presence of outliers. In *2025 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pages 328–336. IEEE, 2025.

[41] P. J. Campo and M. Morari. Robust model predictive control. In *1987 American control conference*, pages 1021–1026. IEEE, 1987.

[42] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE, 2017.

[43] J. S. O. Ceron and P. S. Castro. Revisiting rainbow: Promoting more insightful and inclusive deep reinforcement learning research. In *International Conference on Machine Learning*, pages 1373–1383. PMLR, 2021.

[44] K. H. Chan, A. Mustapha, and M. A. Jubair. Comparative analysis of loss functions in TD3 for autonomous parking. *Journal of Soft Computing and Data Mining*, 5(1):1–14, 2024.

[45] G. Chang. Robust Kalman filtering based on Mahalanobis distance as outlier judging criterion. *Journal of Geodesy*, 88(4):391–401, 2014.

[46] L. Chang, B. Hu, G. Chang, and A. Li. Huber-based novel robust unscented Kalman filter. *IET Science, Measurement & Technology*, 6(6):502–509, 2012.

[47] L. Chang, K. Li, and B. Hu. Huber's M-estimation-based process uncertainty robust filter for integrated INS/GPS. *IEEE Sensors Journal*, 15(6):3367–3374, 2015.

[48] T. Chang, K. Chen, and A. Mehta. Resilient and consistent multirobot cooperative localization with covariance intersection. *IEEE Transactions on Robotics*, 38(1):197–208, 2021.

[49] B. Chen, X. Liu, H. Zhao, and J. C. Principe. Maximum correntropy Kalman filter. *Automatica*, 76:70–77, 2017.

[50] Q. Chen, B. Yu, C. Zhong, Z. Jiang, D. You, and Y. Cai. Study on vehicle state and tire–road friction coefficient estimation based on maximum correntropy generalized high-degree cubature Kalman filter. *Transactions of the Institute of Measurement and Control*, 1–12, 2024.

[51] W. Chen, L. Chen, R. Wang, and M. Pollefeys. LEAP-VO: Long-term effective any point tracking for visual odometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19844–19853, 2024.

[52] W. Chen, Z. Li, Z. Chen, Y. Sun, and Y. Liu. Multiple similarity measure-based maximum correntropy criterion Kalman filter with adaptive kernel width for GPS/INS integration navigation. *Measurement*, 222:113666, 2023.

[53] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun. Monocular 3d object detection for autonomous driving. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2147–2156, 2016.

[54] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1907–1915, 2017.

[55] Y. Chen, W. Li, and Y. Du. A novel robust adaptive Kalman filter with application to urban vehicle integrated navigation systems. *Measurement*, 236:114844, 2024.

[56] Y. Chen, L. Su, and J. Xu. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 1(2):1–25, 2017.

[57] Y. Chen, Y. Sun, J. Li, J. He, and C. He. Tube-based robust model predictive control for autonomous vehicle with complex road scenarios. *Applied Sciences*, 15(12):6471, 2025.

[58] Y. Chen, X. Zhang, K. Zhang, M. Wang, and X. Zhu. Byzantine-robust online and offline distributed reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pages 3230–3269. PMLR, 2023.

[59] D. Chetverikov and D. Stepanov. Robust euclidean alignment of 3d point sets. In *First Hungarian Conference on Computer Graphics and Geometry*, pages 70–75, 2002.

[60] W. T. Chor, C. P. Tan, A. Bakibillah, Z. Pu, and J. Y. Loo. Robust vehicle mass estimation using recursive least M-squares algorithm for intelligent vehicles. *IEEE Transactions on Intelligent Vehicles*, 2023.

[61] S. Chu, H. Qian, S. Yan, and P. Ding. Adaptive robust maximum correntropy cubature Kalman filter for spacecraft attitude estimation. *Advances in Space Research*, 72(8):3376–3385, 2023.

[62] A. I. Comport, E. Malis, and P. Rives. Accurate quadrifocal tracking for robust 3d visual odometry. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 40–45. IEEE, 2007.

[63] A. I. Comport, E. Malis, and P. Rives. Real-time quadrifocal visual odometry. *The International Journal of Robotics Research*, 29(2-3):245–266, 2010.

[64] O. G. Crespillo, A. Andreetti, and A. Grosch. Design and evaluation of robust M-estimators for GNSS positioning in urban environments. In *Proceedings of the 2020 International Technical Meeting of The Institute of Navigation, San Diego, CA, USA*, pages 21–24, 2020.

[65] O. G. Crespillo, D. Medina, J. Skaloud, and M. Meurer. Tightly coupled GNSS/INS integration based on robust M-estimators. In *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pages 1554–1561. IEEE, 2018.

[66] B. Cui, W. Chen, D. Weng, J. Wang, X. Wei, and Y. Zhu. Variational resampling-free cubature Kalman filter for GNSS/INS with measurement outlier detection. *Signal Processing*, 237:110036, 2025.

[67] A. Cunillera, R. M. Lentink, N. van Oort, and R. Goverde. Robust train motion model calibration based on M-estimation. *Authorea Preprints*, 2023.

[68] S. Das. *Robust state estimation methods for robotics applications*. PhD thesis, West Virginia University, 2023.

[69] S. Das, C. Kilic, R. Watson, and J. Gross. A comparison of robust Kalman filters for improving wheel-inertial odometry in planetary rovers. In *Proceedings of the 34th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2021)*, pages 2621–2632, 2021.

[70] P. Dasgupta. Divide and repair: Using options to improve performance of imitation learning against adversarial demonstrations. *arXiv preprint arXiv:2306.04581*, 2023.

[71] P. L. Davies and U. Gather. Breakdown and groups. *The Annals of Statistics*, 33(3):977–1035, 2005.

[72] D. De Palma and G. Indiveri. Output outlier robust state estimation. *International Journal of Adaptive Control and Signal Processing*, 31(4):581–607, 2017.

[73] Z. Deng, Y. Yao, B. Deng, and J. Zhang. A robust loss for point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6138–6147, 2021.

[74] Y. Ding, P. Asseman, É. Chaumette, et al. Robust tightly coupled GNSS/INS experimental assessment for autonomous aircraft inspection. In *Proceedings of the 34th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2021)*, pages 3452–3463, 2021.

[75] Y. Ding, P. Chauchat, G. Pages, and P. Asseman. Learning-enhanced adaptive robust GNSS navigation in challenging environments. *IEEE Robotics and Automation Letters*, 7(4):9905–9912, 2022.

[76] Y. Ding, F. Feriol, Y. Watanabe, P. Asseman, G. Pages, and D. Vivet. Adaptive robust-statistics GNSS navigation based on environmental context detection. In *ION ITM 2023*, 2023.

[77] P. D. Domański and M. Ławryńczuk. Impact of MPC embedded performance index on control quality. *IEEE Access*, 9:24787–24795, 2021.

[78] D. L. Donoho and P. J. Huber. The notion of breakdown point. *A Festschrift for Erich L. Lehmann*, pages 157–184, 1983.

[79] A. Dötlinger and R. M. Kennel. Near time-optimal model predictive control using an l_1-norm based cost functional. In *2014 IEEE Energy Conversion Congress and Exposition (ECCE)*, pages 3504–3511. IEEE, 2014.

[80] A. Dubey and A. Pentland. Private and byzantine-proof cooperative decision-making. *arXiv preprint arXiv:2205.14174*, 2022.

[81] S. R. Eftekhari, S. A. Davari, P. Naderi, C. Garcia, and J. Rodriguez. Robust loss minimization for predictive direct torque and flux control of an induction motor with electrical circuit model. *IEEE Transactions on Power Electronics*, 35(5):5417–5426, 2019.

[82] F. El-Hawary and Y. Jing. Robust regression-based EKF for tracking underwater targets. *IEEE journal of oceanic engineering*, 20(1):31–41, 1995.

[83] M. Elsisi, M. Altius, S.-F. Su, and C.-L. Su. Robust Kalman filter for position estimation of automated guided vehicles under cyberattacks. *IEEE Transactions on Instrumentation and Measurement*, 72:1–12, 2023.

[84] J. Fan, X. Lei, X. Chang, J. Miši, V. B. Miši, and Y. Yao. Less is more: A stealthy and efficient adversarial attack method for DRL-based autonomous driving policies. *IEEE Internet of Things Journal*, 2025.

[85] X. Fan, Y. Ma, Z. Dai, W. Jing, C. Tan, and B. K. H. Low. Fault-tolerant federated reinforcement learning with theoretical guarantee. *Advances in Neural Information Processing Systems*, 34:1007–1021, 2021.

[86] Y. Fan, S. Qiao, G. Wang, and H. Zhang. An improved Sage-Husa variational robust adaptive Kalman filter with uncertain noise covariances. *IEEE Sensors Journal*, 2024.

[87] H. Fang, M. A. Haile, and Y. Wang. Robust extended Kalman filtering for systems with measurement outliers. *IEEE Transactions on Control Systems Technology*, 30(2):795–802, 2021.

[88] Y. Fei, B. Xu, X. Wang, and Y. Guo. A novel current estimation and cooperative localization method under unknown current and non-gaussian noise. *IEEE Transactions on Instrumentation and Measurement*, 2024.

[89] D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Glaeser, F. Timm, W. Wiesbeck, and K. Dietmayer. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1341–1360, 2020.

[90] P. Filzmoser, S. Höppner, I. Ortner, S. Serneels, and T. Verdonck. Cellwise robust M regression. *Computational Statistics & Data Analysis*, 147:106944, 2020.

[91] A. T. Fisch, I. A. Eckley, and P. Fearnhead. Innovative and additive outlier robust Kalman filtering with a robust particle filter. *IEEE Transactions on Signal Processing*, 70:47–56, 2021.

[92] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[93] A. W. Fitzgibbon. Robust registration of 2d and 3d point sets. *Image and vision computing*, 21(13-14):1145–1153, 2003.

[94] A. Fontan, J. Civera, and M. Milford. Adaptive outlier thresholding for bundle adjustment in visual SLAM. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3969–3976. IEEE, 2024.

[95] G. Gagliardi, M. Lupia, G. Cario, and A. Casavola. Optimal H_∞ control for lateral dynamics of autonomous vehicles. *Sensors*, 21(12):4072, 2021.

[96] S. Gaglione, A. Innac, S. P. Carbone, S. Troisi, and A. Angrisano. Robust estimation methods applied to GPS in harsh environments. In *2017 European Navigation Conference (ENC)*, pages 14–25. IEEE, 2017.

[97] M. A. Gandhi and L. Mili. Robust Kalman filter based on a generalized maximum-likelihood-type estimator. *IEEE Transactions on Signal Processing*, 58(5):2509–2520, 2009.

[98] Y. Gao. *Model predictive control for autonomous and semiautonomous vehicles*. PhD thesis, 2014.

[99] J. Garcıa and F. Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.

[100] L. A. García-Escudero, D. Rivera-García, A. Mayo-Iscar, and J. Ortega. Cluster analysis with cellwise trimming and applications for the robust clustering of curves. *Information Sciences*, 573:100–124, 2021.

[101] J. L. García-Lapresta and D. Pérez-Román. Measuring consensus in weak orders. In *Consensual processes*, pages 213–234. Springer, Berlin Heidelberg, 2011.

[102] S. Garg, H. Jönsson, G. Kalander, A. Nilsson, B. Pirange, V. Valadi, and J. Östman. Poisoning attacks on federated learning for autonomous driving. *arXiv preprint arXiv:2405.01073*, 2024.

[103] P. Ge, C. Zhang, T. Zhang, L. Guo, and Q. Xiang. Maximum correntropy square-root cubature Kalman filter with state estimation for distributed drive electric vehicles. *Applied Sciences*, 13(15):8762, 2023.

[104] S. Geisler, T. Schmidt, H. Şirin, D. Zügner, A. Bojchevski, and S. Günnemann. Robustness of graph neural networks at scale. *Advances in Neural Information Processing Systems*, 34:7637–7649, 2021.

[105] S. Geisler, D. Zügner, and S. Günnemann. Reliable graph neural networks via robust aggregation. *Advances in Neural Information Processing Systems*, 33:13272–13284, 2020.

[106] A. Ghorbani, A. Abid, and J. Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3681–3688, 2019.

[107] R. Girod, M. Hauswirth, P. Pfreundschuh, M. Biasio, and R. Siegwart. A robust baro-radar-inertial odometry M-estimator for multicopter navigation in cities and forests. In *2024 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 1–8. IEEE, 2024.

[108] R. Girshick. Fast R-CNN. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[109] R. Gnanadesikan and J. R. Kettenring. Robust estimates, residuals, and outlier detection with multiresponse data. *Biometrics*, pages 81–124, 1972.

[110] T. Gonçalves and A. I. Comport. Real-time direct tracking of color images in the presence of illumination variation. In *2011 IEEE International Conference on Robotics and Automation*, pages 4417–4422. IEEE, 2011.

[111] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[112] L. Gosch, D. Sturm, S. Geisler, and S. Günnemann. Revisiting robustness in graph machine learning. In *Workshop on Trustworthy and Socially Responsible Machine Learning, NeurIPS 2022*.

[113] R. Guerraoui, N. Gupta, and R. Pinot. Byzantine machine learning: A primer. *ACM Computing Surveys*, 56(7):1–39, 2024.

[114] A. Guillard, P. Thevenon, C. Milner, and C. Macabiau. Benefits of CNN-based multipath detection for robust GNSS positioning. In *Proceedings of the 36th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2023)*, pages 283–297, 2023.

[115] J. Guo, L. Ren, X. Zhu, J. Zhuang, B. Jiang, C. Liu, and L. Wang. Pseudo-Huber loss function-based affine registration algorithm of point clouds. In

2024 39th Youth Academic Annual Conference of Chinese Association of Automation (YAC), pages 1034–1039. IEEE, 2024.

[116] M. Guo, C. Bao, Q. Cao, F. Xu, X. Miao, and J. Wu. Fault-tolerant control study of four-wheel independent drive electric vehicles based on drive actuator faults. *Machines*, 12(7):450, 2024.

[117] W. Guo, G. Liu, Z. Zhou, J. Wang, Y. Tang, and M. Wang. Robust training in multiagent deep reinforcement learning against optimal adversary. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2025.

[118] W. Guo, B. Tondi, and M. Barni. An overview of backdoor attacks against deep neural networks and possible defences. *IEEE Open Journal of Signal Processing*, 2022.

[119] S. Gupta, A. Mohanty, and G. Gao. Urban localization using robust filtering at multiple linearization points. *EURASIP Journal on Advances in Signal Processing*, 2023(1):100, 2023.

[120] R. Hable. Asymptotic normality of support vector machine variants and other regularized kernel methods. *Journal of Multivariate Analysis*, 106:92–117, 2012.

[121] F. Hampel, E. Ronchetti, P. Rousseeuw, and W. Stahel. *Robust statistics: The approach based on influence functions*, volume 114. John Wiley & Sons, Hoboken, 2011.

[122] F. R. Hampel. A general qualitative definition of robustness. *The Annals of Mathematical Statistics*, 42(6):1887–1896, 1971.

[123] F. R. Hampel. The influence curve and its role in robust estimation. *Journal of the American Statistical Association*, 69(346):383–393, 1974.

[124] G. Han, F. Liu, J. Deng, W. Bai, X. Deng, and K. Li. An adaptive vehicle tracking enhancement algorithm based on fuzzy interacting multiple model robust cubature Kalman filtering. *Circuits, Systems, and Signal Processing*, 43(1):191–223, 2024.

[125] S.-J. Han, J. Kang, K.-W. Min, and J. Choi. DiLO: Direct light detection and ranging odometry based on spherical range images for autonomous driving. *ETRI journal*, 43(4):603–616, 2021.

[126] P. Hao, O. Karakuş, and A. Achim Robust Kalman filters based on the sub-Gaussian α-stable distribution. *Signal Processing*, 224:1–12, 2024.

[127] B. He, L. Zheng, Y. Jin, and Y. Li. A robust adaptive estimator for sideslip angle and tire-road forces under time-varying and abnormal noise. *IEEE Sensors Journal*, 2025.

[128] M. He, C. Zhu, Q. Huang, B. Ren, and J. Liu. A review of monocular visual odometry. *The Visual Computer*, 36(5): 1053–1065, 2020.

[129] R. He, Y. Shan, and K. Huang. Robust cooperative localization with failed communication and biased measurements. *IEEE Robotics and Automation Letters*, 9(3):2997–3004, 2024.

[130] S. He, S. Han, and F. Miao. Robust electric vehicle balancing of autonomous mobility-on-demand system: A multi-agent reinforcement learning approach. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5471–5478. IEEE, 2023.

[131] S. He, Y. Wang, S. Han, S. Zou, and F. Miao. A robust and constrained multi-agent reinforcement learning electric vehicle rebalancing method in AMoD systems. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5637–5644. IEEE, 2023.

[132] S. He, Z. Zhang, S. Han, L. Pepin, G. Wang, D. Zhang, J. A. Stankovic, and F. Miao. Data-driven distributionally robust electric vehicle balancing for autonomous mobility-on-demand systems under demand and supply uncertainties. *IEEE Transactions on Intelligent Transportation Systems*, 24(5):5199–5215, 2023.

[133] X. He, W. Huang, and C. Lv. Trustworthy autonomous driving via defense-aware robust reinforcement learning against worst-case observational perturbations. *Transportation Research Part C: Emerging Technologies*, 163:104632, 2024.

[134] X. He and C. Lv. Towards safe autonomous driving: Decision making with observation-robust reinforcement learning. *Automotive Innovation*, 6(4):509–520, 2023.

[135] D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

[136] C. Hennig. Dissolution point and isolation robustness: robustness criteria for general cluster analysis methods. *Journal of multivariate analysis*, 99(6):1154–1176, 2008.

[137] S. Hima, B. Lusseti, B. Vanholme, S. Glaser, and S. Mammar. Trajectory tracking for highly automated passenger vehicles. *IFAC Proceedings Volumes*, 44(1):12958–12963, 2011.

[138] J. Ho and S. Ermon. Generative adversarial imitation learning. In *Advances in neural information processing systems*, pages 4565–4573, 2016.

[139] J. Hou, H. He, Y. Yang, T. Gao, and Y. Zhang. A variational Bayesian and Huber-based robust square root cubature Kalman filter for lithium-ion battery state of charge estimation. *Energies*, 12(9):1717, 2019.

[140] E. Hu and L. Sun. Vodrac: Efficient and robust correspondence-based point cloud registration with extreme outlier ratios. *Journal of King Saud University-Computer and Information Sciences*, 35(1):38–55, 2023.

[141] G. Hu, K. Khosoussi, and S. Huang. Towards a reliable SLAM backend. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 37–43. IEEE, 2013.

[142] K. Hu and K. Cheng. Robust tube-based model predictive control for autonomous vehicle path tracking. *IEEE Access*, 10:134389–134403, 2022.

[143] F. Huang, W. Wen, J. Zhang, C. Wang, and L.-T. Hsu. Dynamic object-aware LiDAR odometry aided by joint weightings estimation in urban areas. *IEEE Transactions on Intelligent Vehicles*, 2023.

[144] J. Huang, T. Birdal, Z. Gojcic, L. J. Guibas, and S.-M. Hu. Multiway non-rigid point cloud registration via learned functional map synchronization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[145] T. Huang, L. Peng, R. Vidal, and Y.-H. Liu. Scalable 3d registration via truncated entry-wise absolute residuals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27477–27487, 2024.

[146] Y. Huang, M. Bai, Y. Li, Y. Zhang, and J. Chambers. An improved variational adaptive Kalman filter for cooperative localization. *IEEE Sensors Journal*, 21(9):10775–10786, 2021.

[147] Y. Huang, Y. Zhang, N. Li, and J. Chambers. A robust gaussian approximate fixed-interval smoother for nonlinear systems with heavy-tailed process and measurement noises. *IEEE Signal Processing Letters*, 23(4):468–472, 2016.

[148] Y. Huang, Y. Zhang, P. Shi, and J. Chambers. Variational adaptive Kalman filter with Gaussian-inverse-Wishart mixture distribution. *IEEE Transactions on Automatic Control*, 66(4):1786–1793, 2020.

[149] Y. Huang, Y. Zhang, P. Shi, Z. Wu, J. Qian, and J. A. Chambers. Robust Kalman filters based on gaussian scale mixture distributions with application to target tracking. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 49(10):2082–2096, 2017.

[150] Y. Huang, Y. Zhang, Z. Wu, N. Li, and J. Chambers. A novel adaptive Kalman filter with inaccurate process and measurement noise covariance matrices. *IEEE transactions on Automatic Control*, 63(2):594–601, 2017.

[151] Y. Huang, Y. Zhang, B. Xu, Z. Wu, and J. Chambers. A new outlier-robust Student's t based gaussian approximate filter for cooperative localization. *IEEE/ASME Transactions on Mechatronics*, 22(5):2380–2386, 2017.

[152] Y. Huang, Y. Zhang, Y. Zhao, and J. A. Chambers. A novel robust Gaussian–Student's t mixture distribution based Kalman filter. *IEEE Transactions on signal Processing*, 67(13):3606–3620, 2019.

[153] P. J. Huber and E. Ronchetti. *Robust Statistics*. Wiley, Hoboken, 2009.

[154] M. Hudnell, T. Price, and J.-M. Frahm. Robust aleatoric modeling for future vehicle localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[155] A. Jaegle, S. Phillips, and K. Daniilidis. Fast, robust, continuous monocular egomotion computation. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 773–780. IEEE, 2016.

[156] A. Jeddi, M. J. Shafiee, and A. Wong. A simple fine-tuning is all you need: Towards robust deep learning via adversarial fine-tuning. *arXiv preprint arXiv:2012.13628*, 2020.

[157] D. Jeong and S. B. Choi. Tube-based robust model predictive control for tracking control of autonomous articulated vehicles. *IEEE Transactions on Intelligent Vehicles*, 2023.

[158] Z. Jiang, G. Battistelli, L. Chisci, N. Forti, and W. Zhou. Outlier-robust centralized and distributed variational Bayesian moving horizon estimation. *IEEE Transactions on Signal Processing*, 2025.

[159] X. Jin, Q. Wang, Z. Yan, and H. Yang. Nonlinear robust control of trajectory-following for autonomous ground electric vehicles with active front steering system. *AIMS Math*, 8(5):11151–11179, 2023.

[160] X. Jin, Q. Wang, Z. Yan, H. Yang, and G. Yin. Integrated robust control of path following and lateral stability for autonomous in-wheel-motor-driven electric vehicles. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, page 09544070241227266, 2024.

[161] X. Ju and M. Salibián-Barrera. Robust boosting for regression problems. *Computational Statistics & Data Analysis*, 153:107065, 2021.

[162] S. J. Julier and J. K. Uhlmann. New extension of the Kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI*, volume 3068, pages 182–193. Spie, 1997.

[163] D. Kalaria, Q. Lin, and J. M. Dolan. Delay-aware robust control for safe autonomous driving. In *2022 IEEE Intelligent Vehicles Symposium (IV)*, pages 1565–1571. IEEE, 2022.

[164] R. E. Kalman. A new approach to linear filtering and prediction problems. *Trans. ASME J. Basic Engineering*, 82:34–45, March 1960.

[165] N. Kang, H. Tang, and Y. Han. Robust integrated control of autonomous vehicles path following with response performance improvement considering lateral stability. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, page 09544070241285507, 2024.

[166] C. Karlgaard and H. Schaub. Adaptive Huber-based filtering using projection statistics: Application to spacecraft attitude estimation. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, page 7389, 2008.

[167] C. D. Karlgaard and H. Schaub. Adaptive nonlinear Huber-based navigation for rendezvous in elliptical orbit. *Journal of Guidance, Control, and Dynamics*, 34(2):388–402, 2011.

[168] I. Kazerouni, L. Fitzgerald, G. Dooly, and D. Toal. A survey of state-of-the-art on visual SLAM. *Expert Systems with Applications*, 205: 1–17, 2022.

[169] C. Kerl, J. Sturm, and D. Cremers. Robust odometry estimation for RGB-D cameras. In *2013 IEEE international conference on robotics and automation*, pages 3748–3754. IEEE, 2013.

[170] S. Khalid, N. Rehman, S. Abrar, and L. Mihaylova. Robust Bayesian filtering using Bayesian model averaging and restricted variational In *2018 21st International Conference on Information Fusion (FUSION)*, pages 361–368. IEEE, 2018.

[171] P. Kim, H. Lee, and H. J. Kim. Autonomous flight with robust visual odometry under dynamic lighting conditions. *Autonomous Robots*, 43(6):1605–1622, 2019.

[172] S. Klose, P. Heise, and A. Knoll. Efficient compositional approaches for real-time robust direct visual odometry from RGB-D data. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1100–1106. IEEE, 2013.

[173] N. L. Knight and J. Wang. A comparison of outlier detection procedures and robust estimation methods in GPS positioning. *The Journal of Navigation*, 62(4):699–709, 2009.

[174] P. W. Koh and P. Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pages 1885–1894. PMLR, 2017.

[175] J. Ku, A. D. Pon, and S. L. Waslander. Monocular 3d object detection leveraging accurate proposals and shape reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11867–11876, 2019.

[176] S. Kuutti, S. Fallah, and R. Bowden. Arc: Adversarially robust control policies for autonomous vehicles. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 522–529. IEEE, 2021.

[177] D. Lee, M. Jung, W. Yang, and A. Kim. Lidar odometry survey: recent advancements and remaining challenges. *Intelligent Service Robotics*, 17(2): 95-118, 2024.

[178] J.-K. Lee, Y.-K. Baik, H. Cho, K. Kim, and D. H. Kim. 1-point RANSAC-based method for ground object pose estimation. *arXiv preprint arXiv:2008.03718*, 2020.

[179] T. Lee and Y. Jeong. A tube-based model predictive control for path tracking of autonomous articulated vehicle. In *Actuators*, volume 13, page 164. MDPI, 2024.

[180] A. Leung, V. Yohai, and R. Zamar. Multivariate location and scatter matrix estimation under cellwise and casewise contamination. *Computational Statistics & Data Analysis*, 111:59–76, 2017.

[181] A. Leung, H. Zhang, and R. Zamar. Robust regression estimation and inference in the presence of cellwise and casewise contamination. *Computational Statistics & Data Analysis*, 99:1–11, 2016.

[182] J. Li, Q. Hu, and M. Ai. Point cloud registration based on one-point RANSAC and scale-annealing biweight estimation. *IEEE Transactions on Geoscience and Remote Sensing*, 59(11):9716–9729, 2021.

[183] J. Li, L. Tao, W. Zou, Y. Zhang, B. Shuai, J. Duan, S. E. Li, H. Sun, Y. Wang, Y. Gao, et al. Towards robust motion control in multi-source uncertain scenarios by robust policy iteration. *Communications in Transportation Research*, 5:100191, 2025.

[184] Q. Li, Y. Ben, S. M. Naqvi, J. A. Neasham, and J. A. Chambers. Robust student's t-based cooperative navigation for autonomous underwater vehicles. *IEEE Transactions on Instrumentation and Measurement*, 67(8):1762–1777, 2018.

[185] R. Li, H. Xu, and R. Zheng. Cauchy kernel-based maximum correntropy extended Kalman filter for cooperative localization of multi-UUVs. In *Proceedings of the 2024 3rd International Symposium on Intelligent Unmanned Systems and Artificial Intelligence*, pages 290–293, 2024.

[186] S. Li, E. Ngai, and T. Voigt. Byzantine-robust aggregation in federated learning empowered industrial IoT. *IEEE Transactions on Industrial Informatics*, 19(2):1165–1175, 2021.

[187] S. Li, Z. Wang, F. Juefei-Xu, Q. Guo, X. Li, and L. Ma. Common corruption robustness of point cloud detectors: Benchmark and enhancement. *IEEE Transactions on Multimedia*, 2023.

[188] Y. Li, Z. Gao, C. Yang, and Q. Xu. A novel UWB/INS tight integration model based on ranging offset calibration and robust cubature Kalman filter. *Measurement*, 237:115186, 2024.

[189] Y. Li, L. Hou, Y. Yang, and J. Tong. Huber's M-estimation-based cubature Kalman filter for an INS/DVL integrated system. *Mathematical Problems in Engineering*, 2020:1–12, 2020.

[190] Y. Li, J. Li, J. Qi, and L. Chen. Robust cubature Kalman filter for dynamic state estimation of synchronous machines under unknown measurement noise statistics. *IEEE Access*, 7:29139–29148, 2019.

[191] Y. Li and L. Zhang. A robust Kalman filter based on Gaussian-Exponential distribution for SINS/USBL integration navigation system. *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, page 14750902231224832, 2024.

[192] H. Lim, K. Han, D. Kim, G. Shin, and G. Kim, S. Hong, and H. Myung. Orora: Outlier-robust radar odometry. *arXiv preprint arXiv:2303.01876*, 2023.

[193] H. Lim, B. Kim, D. Kim, E. Mason Lee, and H. Myung. Quatro++: Robust global registration exploiting ground segmentation for loop closing in lidar slam. *The International Journal of Robotics Research*, 43(5):685–715, 2024.

[194] L. Lin, X. Zhang, and Z. Xiao. Integration of SINS, laser doppler velocimeter, and monocular visual odometry for autonomous navigation in complex road environments. *Optik*, 295:171513, 2023.

[195] J. Liu, Q. Kong, F. Yin, Z. Cai, M. Sun, and B. Chen. Design and implementation of the correntropy-based filter for GNSS vector tracking and positioning. *IEEE Transactions on Aerospace and Electronic Systems*, 2025.

[196] L. Liu, W. Bi, B. Zhang, Z. Huang, A. Zhang, and S. Xu. TORKF: A dual-driven Kalman filter for outlier-robust state estimation and application to aircraft tracking. *Aerospace*, 12(8):660, 2025.

[197] L. Liu, R. Mu, and N. Cui. Huber-based robust tracking method of hypersonic cruise vehicle. In *China Aeronautical Science and Technology Conference*, pages 484–501. Springer, 2023.

[198] L. Liu, Z. Tang, L. Li, and D. Luo. Robust imitation learning from corrupted demonstrations. *arXiv preprint arXiv:2201.12594*, 2022.

[199] W. Liu, G. Chen, and A. Knoll. Matrix inequalities based robust model predictive control for vehicle considering model uncertainties, external disturbances, and time-varying delay. *Frontiers in Neurorobotics*, 14:617293, 2021.

[200] X. Liu, X. Liu, Y. Yang, and W. Zhang. An INS/Lidar integrated navigation algorithm based on robust Kalman filter. In *Advances in Guidance, Navigation and Control: Proceedings of 2020 International Conference on Guidance, Navigation and Control, ICGNC 2020, Tianjin, China, October 23–25, 2020*, pages 1027–1037. Springer, 2022.

[201] Y. Liu, D. Cui, and C. Xie. Measurement and estimation of vehicle state based on improved maximum correntropy square-root cubature Kalman filter. *Journal of Measurements in Engineering*, 2025.

[202] Y. Liu, J. Zhang, L. Fang, Q. Jiang, and B. Zhou. Multimodal motion prediction with stacked transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7577–7586, 2021.

[203] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.

[204] D. L. S. Lubanco, A. Hashem, M. Pichler-Scheder, A. Stelzer, R. Feger, and T. Schlechter. R^{3}o: Robust radon radar odometry. *IEEE Transactions on Intelligent Vehicles*, 2023.

[205] L. Luft, T. Schubert, S. Roumeliotis, and W. Burgard. Recursive Decentralized Collaborative Localization for Sparsely Communicating Robots. *Robotics: science and systems*, volume 12, 2016.

[206] R. W. Lutz, M. Kalisch, and P. Bühlmann. Robustified L_2 boosting. *Computational Statistics & Data Analysis*, 52(7):3331–3341, 2008.

[207] C. Ma, S. Pan, W. Gao, H. Wang, and L. Liu. Variational Bayesian-based robust adaptive filtering for GNSS/INS tightly coupled positioning in urban environments. *Measurement*, 223:113668, 2023.

[208] E. Ma, R. Etesami, et al. Local environment poisoning attacks on federated reinforcement learning. *arXiv preprint arXiv:2303.02725*, 2023.

[209] T. Ma, R. Zhang, S. Gao, H. Li, and Y. Zhang. A variational Bayesian truncated adaptive filter for uncertain systems with inequality constraints. *IET Signal Processing*, 2024(1):3809689, 2024.

[210] K. MacTavish and T. D. Barfoot. At all costs: A comparison of robust cost functions for camera correspondence outliers. In *2015 12th conference on computer and robot vision*, pages 62–69. IEEE, 2015.

[211] W. Maddern, A. Stewart, C. McManus, B. Upcroft, W. Churchill, and P. Newman. Illumination invariant imaging: Applications in robust vision-based localisation, mapping and classification for autonomous vehicles. In *Proceedings of the Visual Place Recognition in Changing Environments Workshop, IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China*, volume 2, page 5, 2014.

[212] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[213] R. A. Maronna, R. D. Martin, V. J. Yohai, and M. Salibián-Barrera. *Robust statistics: theory and methods (with R)*. John Wiley & Sons, Hoboken, 2019.

[214] T. Masuda, K. Sakaue, and N. Yokoya. Registration and integration of multiple range images for 3-d model construction. In *Proceedings of 13th international conference on pattern recognition*, volume 1, pages 879–883. IEEE, 1996.

[215] S. Mata, A. Zubizarreta, and C. Pinto. Robust tube-based model predictive control for lateral path tracking. *IEEE Transactions on Intelligent Vehicles*, 4(4):569–577, 2019.

[216] D. Medina. *Robust GNSS Carrier Phase-based Position and Attitude Estimation*. PhD thesis, Universidad Carlos III de Madrid, 2022.

[217] D. Medina, H. Li, J. Vila-Valls, and P. Closas. On robust statistics for GNSS single point positioning. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 3281–3287. IEEE, 2019.

[218] D. Medina, H. Li, J. Vilà-Valls, and P. Closas. Robust statistics for GNSS positioning under harsh conditions: A useful tool? *Sensors*, 19(24):5402, 2019.

[219] M. Meilland, A. I. Comport, and P. Rives. Real-time dense visual tracking under large lighting variations. In *British Machine Vision Conference*, pages 45–1. British Machine Vision Association, 2011.

[220] F. Memarian, A. Hashemi, S. Niekum, and U. Topcu. Robust generative adversarial imitation learning via local lipschitzness. *arXiv preprint arXiv:2107.00116*, 2021.

[221] D. Ming, X. Wu, Y. Wang, Z. Zhu, H. Ge, and R. Liu. A real-time monocular visual SLAM based on the bundle adjustment with adaptive robust kernel. *Journal of Intelligent & Robotic Systems*, 107(3):35, 2023.

[222] U. Montanaro, S. Dixit, and S. Fallah. Adaptive control and robust MPC for linearising longitudinal vehicle dynamics for platooning applications. In *Advances in Dynamics of Vehicles on Roads and Tracks: Proceedings of the 26th Symposium of the International Association of Vehicle System Dynamics, IAVSD 2019, August 12-16, 2019, Gothenburg, Sweden*, pages 1052–1061. Springer, 2020.

[223] S. R. Mousa, S. Ishak, R. M. Mousa, J. Codjoe, and M. Elhenawy. Deep reinforcement learning agent with varying actions strategy for solving the eco-approach and departure problem at signalized intersections. *Transportation research record*, 2674(8):119–131, 2020.

[224] J. Murrugarra-Llerena, L. N. Kirsten, and C. R. Jung. Can we trust bounding box annotations for object detection? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4813–4822, 2022.

[225] R. Muthukrishnan, E. Boobalan, and R. Reka. Performance of RANSAC techniques under classical and robust methods. *IJIRCCE*, 2:1–5, 2014.

[226] J. M. Nadales, A. Hakobyan, D. M. de la Peña, D. Limon, and I. Yang. Risk-aware Wasserstein distributionally robust control of vessels in natural waterways. *IEEE Transactions on Control Systems Technology*, 2024.

[227] R. Nebeluk and M. Ławryńczuk. Nonlinear model predictive control with l_1 cost-function using neural networks for multivariable processes. *IFAC-PapersOnLine*, 56(2):1591–1596, 2023.

[228] Y. Nemmour, B. Schölkopf, and J.-J. Zhu. Approximate distributionally robust nonlinear optimization with application to model predictive control: A functional approach. In *Learning for Dynamics and Control*, pages 1255–1269. PMLR, 2021.

[229] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. DTAM: Dense tracking and mapping in real-time. In *2011 international conference on computer vision*, pages 2320–2327. IEEE, 2011.

[230] M. Nezami, H. S. Abbas, N. T. Nguyen, and G. Schildbach. Robust tube-based LPV-MPC for autonomous lane keeping. *IFAC-PapersOnLine*, 55(35):103–108, 2022.

[231] H. T. Nguyen and A. W. Smeulders. Fast occluded object tracking by a robust appearance filter. *IEEE transactions on pattern analysis and machine intelligence*, 26(8):1099–1104, 2004.

[232] A. Nilim and L. El Ghaoui. Robust control of Markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

[233] H. Nurminen, T. Ardeshiri, R. Piche, and F. Gustafsson. Robust inference for state-space models with skewed measurement noise. *IEEE Signal Processing Letters*, 22(11):1898–1902, 2015.

[234] A. Nurunnabi, F. Teferle, R. Lindenbergh, J. Li, and S. Zlatanova. Robust approach for urban road surface extraction using mobile laser scanning 3d point clouds. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 43(B1-2022), 2022.

[235] A. Nurunnabi, G. West, and D. Belton. Robust locally weighted regression techniques for ground surface points filtering in mobile laser scanning three dimensional point cloud data. *IEEE Transactions on Geoscience and Remote Sensing*, 54(4):2181–2193, 2015.

[236] F. Oudjama, A. Boumediene, K. Saidi, and D. Boubekeur. Robust speed control in nonlinear electric vehicles using H-infinity control and the LMI approach. *J. Intell Syst. Control*, 2(3):170–182, 2023.

[237] C.-H. Park and J.-H. Chang. Robust range estimation algorithm based on hyper-tangent loss function. *IET Signal Processing*, 14(5):314–321, 2020.

[238] S. Park, T. Schöps, and M. Pollefeys. Illumination change robustness in direct visual SLAM. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 4523–4530. IEEE, 2017.

[239] A. Parra Bustos and T.-J. Chin. Guaranteed outlier removal for rotation search. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2165–2173, 2015.

[240] A. Patterson, V. Liao, and M. White. Robust losses for learning value functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):6157–6167, 2022.

[241] S. Paul, B. Jhamb, D. Mishra, and M. S. Kumar. Edge loss functions for deep-learning depth-map. *Machine Learning with Applications*, 7:100218, 2022.

[242] D. Penco, J. Davins-Valldaura, E. Godoy, P. Kvieska, and G. Valmorbida. Self-scheduled H_∞ control of autonomous vehicle in collision avoidance maneuvers. *IFAC-PapersOnLine*, 54(8):148–153, 2021.

[243] L. Peng, C. Kümmerle, and R. Vidal. On the convergence of IRLS and its variants in outlier-robust estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17808–17818, 2023.

[244] K. Pereida and A. P. Schoellig. All models are wrong: Robust adaptive model predictive control for safe and high accuracy trajectory tracking in the presence of model errors. 2019.

[245] J. M. Phillips, R. Liu, and C. Tomasi. Outlier robust ICP for minimizing fractional RMSD. In *Sixth International Conference on 3-D Digital Imaging and Modeling (3DIM 2007)*, pages 427–434. IEEE, 2007.

[246] R. Piché, S. Särkkä, and J. Hartikainen. Recursive outlier-robust filtering and smoothing for nonlinear systems using the multivariate Student-t distribution. In *2012 IEEE International Workshop on Machine Learning for Signal Processing*, pages 1–6. IEEE, 2012.

[247] L. Pinto, J. Davidson, R. Sukthankar, and A. Gupta. Robust adversarial reinforcement learning. In *International conference on machine learning*, pages 2817–2826. PMLR, 2017.

[248] D. Qi, J. Feng, Y. Li, L. Wang, and B. Song. A robust hierarchical estimation scheme for vehicle state based on maximum correntropy square-root cubature Kalman filter. *Entropy*, 25(3):453, 2023.

[249] W. Qi, W. Qin, and Z. Yun. Closed-loop state of charge estimation of Li-ion batteries based on deep learning and robust adaptive Kalman filter. *Energy*, 307:132805, 2024.

[250] Q. Qiang, L. Baojun, L. Yingchun, L. Xia, and W. Shen. Robust UKF orbit determination method with time-varying forgetting factor for angle/range-based integrated navigation system. *Chinese Journal of Aeronautics*, 37(11):420–434, 2024.

[251] T. Qin, P. Li, and S. Shen. VINS-Mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics*, 34(4):1004–1020, 2018.

[252] Y. Qu, R. Ma, and Z. Gao. Heavy-tailed filtering with reduced sensitivity to inaccurate noise covariance. *Signal Processing*, page 110209, 2025.

[253] R. C. Rafaila and G. Livint. H-infinity control of automatic vehicle steering. In *2016 International Conference and Exposition on Electrical and Power Engineering (EPE)*, pages 031–036. IEEE, 2016.

[254] R. Ranftl and V. Koltun. Deep fundamental matrix estimation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 284–299, 2018.

[255] J. Raymaekers and P. J. Rousseeuw. Handling cellwise outliers by sparse regression and robust covariance. *arXiv preprint arXiv:1912.12446*, 2019.

[256] Y. Ren, J. Duan, S. E. Li, Y. Guan, and Q. Sun. Improving generalization of reinforcement learning with minimax distributional soft actor-critic. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6. IEEE, 2020.

[257] Y. Ren, G. Zhan, L. Tang, S. E. Li, J. Jiang, K. Li, and J. Duan. Improve generalization of driving policy at signalized intersections with adversarial learning. *Transportation Research Part C: Emerging Technologies*, 152:104161, 2023.

[258] S. Rhode. *Robust and regularized algorithms for vehicle tractive force prediction and mass estimation*, volume 62. KIT Scientific Publishing, Karlsruhe, 2018.

[259] H. Rieder. *Robust asymptotic statistics*, volume 1. Springer Science & Business Media, Berlin, 1994.

[260] H. Rieder, M. Kohl, and P. Ruckdeschel. The cost of not knowing the radius. *Statistical Methods & Applications*, 17(1):13–40, 2008.

[261] F. Roselli, M. Corno, S. M. Savaresi, M. Giorelli, D. Azzolini, A. Irilli, and G. Panzani. H_∞ control with look-ahead for lane keeping in autonomous vehicles. In *2017 IEEE Conference on Control Technology and Applications (CCTA)*, pages 2220–2225. IEEE, 2017.

[262] P. J. Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, 79(388):871–880, 1984.

[263] P. J. Rousseeuw and W. Van Den Bossche. Detecting deviating data cells. *Technometrics*, 60(2):135–145, 2018.

[264] P. J. Rousseeuw and K. Van Driessen. An algorithm for positive-breakdown regression based on concentration steps. In *Data Analysis*, pages 335–346. Springer, Berlin Heidelberg, 2000.

[265] P. J. Rousseeuw and K. Van Driessen. Computing LTS regression for large data sets. *Data mining and knowledge discovery*, 12(1):29–45, 2006.

[266] W. Ruan, X. Yi, and X. Huang. Adversarial robustness of deep learning: Theory, algorithms, and applications. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pages 4866–4869, 2021.

[267] P. Ruckdeschel. *Ansätze zur Robustifizierung des Kalman-Filters*. PhD thesis, University of Bayreuth, 2001.

[268] P. Ruckdeschel and N. Horbenko. Yet another breakdown point notion: EFSBP. *Metrika*, 75(8):1025–1047, 2012.

[269] P. Ruckdeschel, B. Spangl, and D. Pupashenko. Robust Kalman tracking and smoothing with propagating and non-propagating outliers. *Statistical Papers*, 55(1):93–123, 2014.

[270] F. Ruwisch and S. Schön. Feature map aided robust high precision GNSS positioning in harsh urban environments. *IEEE Transactions on Intelligent Transportation Systems*, 2025.

[271] A. Saday, İ. Demirel, Y. Yıldırım, and C. Tekin. Federated multi-armed bandits under byzantine attacks. *IEEE Transactions on Artificial Intelligence*, 2025.

[272] A. Samadzadeh and A. Nickabadi. SRVIO: Super robust visual inertial odometry for dynamic environments and challenging loop-closure conditions. *IEEE Transactions on Robotics*, 39(4):2878–2891, 2023.

[273] S. Särkkä and J. Hartikainen. Non-linear noise adaptive Kalman filtering via variational Bayes. In *2013 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE, 2013.

[274] S. Särkkä and A. Nummenmaa. Recursive noise adaptive Kalman filtering by variational Bayesian approximations. *IEEE Transactions on Automatic control*, 54(3):596–600, 2009.

[275] R. Schmied. *Mixed H2/H and predictive adaptive cruise control: design, evaluation and comparison/eingereicht von Roman Schmied*. PhD thesis, Universität Linz, 2017.

[276] P. Schmuck, and M. Chli. CCM-SLAM: Robust and efficient centralized collaborative monocular simultaneous localization and mapping for robotic teams. *Journal of Field Robotics*, 36(4): 763–781, 2019.

[277] D. Schneider. *Optimierte Schätzverfahren für intelligente Batteriesysteme*. PhD thesis, Technische Universität München, 2024.

[278] M. Schuurmans, A. Katriniok, C. Meissen, H. E. Tseng, and P. Patrinos. Safe, learning-based MPC for highway driving under lane-change uncertainty: A distributionally robust approach. *Artificial Intelligence*, 320:103920, 2023.

[279] S. D.-C. Shashua and S. Mannor. Deep robust Kalman filter. *arXiv preprint arXiv:1703.02310*, 2017.

[280] F. Shen, C. Shen, A. van den Hengel, and Z. Tang. Approximate least trimmed sum of squares fitting and applications in image analysis. *IEEE Transactions on Image Processing*, 22(5):1836–1847, 2013.

[281] H. Shen, G. Wen, Y. Lv, and J. Zhou. A stochastic event-triggered robust unscented Kalman filter-based USV parameter estimation. *IEEE Transactions on Industrial Electronics*, 71(9):11272–11282, 2023.

[282] Q. Shen, Y. Li, H. Jiang, Z. Wang, and T. Zhao. Deep reinforcement learning with robust and smooth policy. In *International Conference on Machine Learning*, pages 8707–8718. PMLR, 2020.

[283] J. Shi, H. Yang, and L. Carlone. Optimal and robust category-level perception: Object pose and shape estimation from 2-d and 3-d semantic keypoints. *IEEE Transactions on Robotics*, 2023.

[284] K. Si, P. Li, Z.-p. Yuan, K. Qiao, B. Wang, and X. He. Distributionally robust Kalman filtering for INS/GPS tightly coupled integration with model uncertainty and measurement outlier. *IEEE Transactions on Instrumentation and Measurement*, 2023.

[285] D. Simon. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, Hoboken, 2006.

[286] A. P. Singh and R. Tali. Byzantine resilient federated reinforce (gm-fedreinforce). In *2023 International Conference on Machine Learning and Applications (ICMLA)*, pages 825–830. IEEE, 2023.

[287] R. Soloperto, J. Köhler, F. Allgöwer, and M. A. Müller. Collision avoidance for uncertain nonlinear systems with moving obstacles using robust model predictive control. In *2019 18th European Control Conference (ECC)*, pages 811–817. IEEE, 2019.

[288] H. W. Sorenson. Least-squares estimation: from Gauss to Kalman. *IEEE spectrum*, 7(7):63–68, 1970.

[289] V.-J. Štironja, L. Petrović, J. Peršić, I. Marković, and I. Petrović. RAVE: A framework for radar ego-velocity estimation. In *2024 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 1–6. IEEE, 2024.

[290] H. Sun, L. Dai, and P. Wang. An efficient moving obstacle avoidance scheme for UAVs via output feedback robust MPC. *IEEE Transactions on Aerospace and Electronic Systems*, 60(5):6199–6212, 2024.

[291] L. Sun. Ransic: Fast and highly robust estimation for rotation search and point cloud registration using invariant compatibility. *IEEE Robotics and Automation Letters*, 7(1):143–150, 2021.

[292] W. Sun, J. Zhao, W. Ding, and P. Sun. Robust UKF relative positioning approach for tightly coupled vehicle ad hoc networks based on adaptive M-estimation. *IEEE Sensors Journal*, 23(9):9959–9971, 2023.

[293] X. Sun, P. Li, J. Zhang, Z. Chen, and B. He. Robust AUV navigation with non-gaussian noise: Enhanced UKF with maximum correntropy and M-estimation methods. *Robotics and Autonomous Systems*, page 105007, 2025.

[294] Z. Sun, W. Gao, X. Tao, S. Pan, P. Wu, and H. Huang. Semi-tightly coupled robust model for GNSS/UWB/INS integrated positioning in challenging environments. *Remote Sensing*, 16(12):2108, 2024.

[295] R. S. Sutton, D. Precup, and S. Singh. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2):181–211, 1999.

[296] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[297] Y. Tang, M. Wang, Y. Yang, Z. Lan, and Y. Yue. Robust large-scale collaborative localization based on semantic submaps with extreme outliers. *IEEE/ASME Transactions on Mechatronics*, 29(4):2649–2660, 2023.

[298] Z. Tang. *Robust imitation learning from observation*. PhD thesis, 2020.

[299] Distributed Fault-Tolerant Multi-Robot Cooperative Localization in Adversarial Environments. *arXiv preprint arXiv:2507.06750*, 2025.

[300] A. T. Thorgeirsson, S. Scheubner, S. Fünfgeld, and F. Gauterin. An investigation into key influence factors for the everyday usability of electric vehicles. *IEEE Open Journal of Vehicular Technology*, 1:348–361, 2020.

[301] B. Thörnblom. Predicting impact of training data for pedestrian detection in autonomous vehicles using influence functions. 2019.

[302] C. E. Thorpe and T. Kanade. *Second Annual Report for Perception for Outdoor Navigation*. Carnegie Mellon University, The Robotics Institute, Pittsburgh, Pennsylvania, 1991.

[303] Y. Tian, Y. Chang, F. H. Arias, C. Nieto-Granda, J. P. How, and L. Carlone. Kimera-multi: Robust, distributed, dense metric-semantic SLAM for multi-robot systems. *IEEE Transactions on Robotics*, 38(4), 2022.

[304] S. Truzman, G. Revach, N. Shlezinger, and I. Klein. Outlier-insensitive Kalman filtering using NUV priors. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[305] S. Truzman, G. Revach, N. Shlezinger, and I. Klein. Outlier-insensitive Kalman filtering: Theory and applications. *IEEE Sensors Journal*, 2024.

[306] D. Tschirschwitz, C. Benz, M. Florek, H. Norderhus, B. Stein, and V. Rodehorst. Drawing the same bounding box twice? Coping noisy annotations in object detection with repeated labels. In *DAGM German Conference on Pattern Recognition*, pages 605–623. Springer, 2023.

[307] C.-H. Tseng, S.-F. Lin, and D.-J. Jwo. Robust Huber-based cubature Kalman filter for GPS navigation processing. *The Journal of Navigation*, 70(3):527–546, 2017.

[308] J. Tu, M. Ren, S. Manivasagam, M. Liang, B. Yang, R. Du, F. Cheng, and R. Urtasun. Physically realizable adversarial examples for LiDAR object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13716–13725, 2020.

[309] M. P. Uddin, Y. Xiang, M. Hasan, J. Bai, Y. Zhao, and L. Gao. A systematic literature review of robust federated learning: Issues, solutions, and future research directions. *ACM Computing Surveys*, 57(10):1–62, 2025.

[310] H. Velasco, H. Laniado, M. Toro, V. Leiva, and Y. Lio. Robust three-step regression based on comedian and its performance in cell-wise and case-wise outliers. *Mathematics*, 8(8):1259, 2020.

[311] R. Von Mises. On the asymptotic distribution of differentiable statistical functions. *The Annals of Mathematical Statistics*, 18(3):309–348, 1947.

[312] E. A. Wan and R. Van Der Merwe. The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 adaptive systems for signal processing, communications, and control symposium (Cat. No. 00EX373)*, pages 153–158. Ieee, 2000.

[313] W. Wan, J. Feng, B. Song, and X. Li. Huber-based robust unscented Kalman filter distributed drive electric vehicle state observation. *Energies*, 14(3):750, 2021.

[314] G. Wang, Z. Zhang, C. Yang, L. Ma, and W. Dai. Robust EKF based on shape parameter mixture distribution for wireless localization with time-varying skewness measurement noise. *IEEE Transactions on Instrumentation and Measurement*, 2024.

[315] J. Wang, L.-j. Qian, J. Chen, L. Xuan, and X. Chen. Multi-innovation adaptive UKF with robust estimation using QS decomposition for vehicle state estimation. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, page 09544070241313090, 2025.

[316] J. Wang, Z. Zhuang, Y. Wang, and H. Zhao. Adversarially robust imitation learning. In *Conference on Robot Learning*, pages 320–331. PMLR, 2022.

[317] L. Wang, H. Chen, F. Lian, W. Zhang, and J. Liu. Robust Bayesian recursive ensemble Kalman filter under the non-stationary heavy-tailed noise. *IEEE Sensors Journal*, 2024.

[318] R. Wang, D. Becker, and T. Hobiger. Stochastic modeling with robust Kalman filter for real-time kinematic GPS single-frequency positioning. *GPS Solutions*, 27(3):153, 2023.

[319] S. Wang, Q. Li, Z. Cui, J. Hou, and C. Huang. Bandit-based data poisoning attack against federated learning for autonomous driving models. *Expert Systems with Applications*, 227:120295, 2023.

[320] S. Wang, Y. Zhang, and H. Li. Satellite image based cross-view localization for autonomous vehicle. *arXiv preprint arXiv:2207.13506*, 2022.

[321] X. Wang, H. Zhang, A. Bilal, H. Long, and X. Liu. WGM-dSAGA: Federated learning strategies with byzantine robustness based on weighted geometric median. *Electronics*, 12(5):1190, 2023.

[322] Y. Wang, S. Sun, and L. Li. Adaptively robust unscented Kalman filter for tracking a maneuvering vehicle. *Journal of Guidance, Control, and Dynamics*, 37(5):1696–1701, 2014.

[323] Y. Wang, Q. Yu, and Y. Shen. Robust Bayesian cooperative positioning for intelligent vehicles using GNSS and V2V range measurements. *IEEE Transactions on Wireless Communications*, 2025.

[324] T. Werner. Asymptotic linear expansion of regularized M-estimators. *Annals of the Institute of Statistical Mathematics*, 74(1):167–194, 2022.

[325] T. Werner. Quantitative robustness of instance ranking problems. *Annals of the Institute of Statistical Mathematics*, pages 1–34, 2022.

[326] T. Werner. Trimming stability selection increases variable selection robustness. *Machine Learning*, 112(12):4995–5055, 2023.

[327] T. Werner. Global quantitative robustness of regression feed-forward neural networks. *Neural Computing and Applications*, (36):19967–19988, 2024.

[328] M. West. Robust sequential approximate Bayesian estimation. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 43(2):157–166, 1981.

[329] A. Wischnewski. *Robust and data-driven control for autonomous racing*. PhD thesis, Technische Universität München, 2023.

[330] D. Withers and P. Newman. Modelling scene change for large-scale long term laser localisation. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6233–6239. IEEE, 2017.

[331] J. Wörmann, D. Bogdoll, E. Bührle, H. Chen, E. F. Chuo, K. Cvejoski, L. van Elst, T. Gleißner, P. Gottschall, S. Griesche, et al. Knowledge augmented machine learning with applications in autonomous driving: A survey. *arXiv preprint arXiv:2205.04712*, 2022.

[332] F. Wu, C. Sun, H. Li, and S. Zheng. Real-time center of gravity estimation for intelligent connected vehicle based on HEKF-EKF. *Electronics*, 12(2):386, 2023.

[333] P. Wu, S. Chen, and D. N. Metaxas. Motionnet: Joint perception and motion prediction for autonomous driving based on bird's eye view maps. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11385–11395, 2020.

[334] Q. Wu, L. Chen, Y. Li, Z. Wang, S. Yao, and H. Li. Reweighted robust particle filtering approach for target tracking in automotive radar application. *Remote Sensing*, 14(21):5477, 2022.

[335] W. Wu, J. Li, C. Chen, B. Yang, X. Zou, Y. Yang, Y. Xu, R. Zhong, and R. Chen. AFLI-Calib: Robust LiDAR-IMU extrinsic self-calibration based on adaptive frame length LiDAR odometry. *ISPRS Journal of Photogrammetry and Remote Sensing*, 199:157–181, 2023.

[336] M. Xia, T. Zhang, L. Zhang, B. Yang, and Y. Shi. A mixture distribution-based robust SINS/USBL integration navigation with time-varying delays. *IEEE Transactions on Instrumentation and Measurement*, 2024.

[337] L. Xiong, J. Shen, and X. Bi. A Huber based unscented Kalman filter terrain matching algorithm for underwater autonomous vehicle. In *Proceedings of the 3rd International Conference on Computer Science and Application Engineering*, pages 1–8, 2019.

[338] P. Xiong, M. Tegegn, J. S. Sarin, S. Pal, and J. Rubin. It is all about data: A survey on the effects of data on adversarial robustness. *ACM Computing Surveys*, 56(7):1–41, 2024.

[339] X. Xiong, W. Chen, Z. Liu, and Q. Shen. DS-VIO: Robust and efficient stereo visual inertial odometry based on dual stage ekf. *arXiv preprint arXiv:1905.00684*, 2019.

[340] L. Xu, W. Zhuang, G. Yin, G. Li, and C. Bian. Robust overtaking control of autonomous electric vehicle with parameter uncertainties. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of automobile engineering*, 233(13):3358–3376, 2019.

[341] N. Xu, R. Qin, and S. Song. Point cloud registration for LiDAR and photogrammetric data: A critical synthesis and performance analysis on classic and deep learning algorithms. *ISPRS Open Journal of Photogrammetry and Remote Sensing*, page 100032, 2023.

[342] Y. Xu, J. Shao, S. Gao, and Y. Zhou. Emergency autonomous steering control research based on receding horizon H_∞ control and minimax criteria. *IEEE Access*, 9:151781–151792, 2021.

[343] F. Yan, Z. Li, and Z. Zhou. Robust and efficient edge-based visual odometry. *Computational Visual Media*, 8(3):467–481, 2022.

[344] J. Yan, F. Zhu, Y. Huang, and Y. Zhang. Robust decentralized cooperative localization for multirobot system against measurement outliers. *IEEE Internet of Things Journal*, 11(10):17934–17947, 2024.

[345] Q. Yan, J. Chen, and L. De Strycker. An outlier detection method based on Mahalanobis distance for source localization. *Sensors*, 18(7):2186, 2018.

[346] H. Yang, P. Antonante, V. Tzoumas, and L. Carlone. Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection. *IEEE Robotics and Automation Letters*, 5(2):1127–1134, 2020.

[347] H. Yang and L. Carlone. One ring to rule them all: Certifiably robust geometric perception with outliers. *Advances in neural information processing systems*, 33:18846–18859, 2020.

[348] H. Yang and L. Carlone. Certifiably optimal outlier-robust geometric perception: Semidefinite relaxations and scalable global optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):2816–2834, 2022.

[349] S.-W. Yang and C.-C. Wang. Simultaneous egomotion estimation, segmentation, and moving object detection. *Journal of Field Robotics*, 28(4):565–588, 2011.

[350] X. Yang, F. Wu, L. Gui, and S. Zhong. A tube-based model predictive control method for intelligent vehicles path tracking. *Cluster Computing*, 27(8):10343–10357, 2024.

[351] D. Yin, Y. Chen, R. Kannan, and P. Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International Conference on Machine Learning*, pages 5650–5659. PMLR, 2018.

[352] Z. Yin, J. Yang, Y. Ma, S. Wang, D. Chai, and H. Cui. A robust adaptive extended Kalman filter based on an improved measurement noise covariance matrix for the monitoring and isolation of abnormal disturbances in GNSS/INS vehicle navigation. *Remote Sensing*, 15(17):4125, 2023.

[353] K. Yousif, A. Bab-Hadiashar, and R. Hoseinnezhad. An overview to visual odometry and visual SLAM: Applications to mobile robotics. *Intelligent Industrial Systems*, 1(4): 289–311, 2015.

[354] A. Yu, R. Palefsky-Smith, and R. Bedi. Deep reinforcement learning for simulated autonomous vehicle control. *Course Project Reports: Winter*, 2016:1–7, 2016.

[355] S. Yu, M. Hirche, Y. Huang, H. Chen, and F. Allgöwer. Model predictive control for autonomous ground vehicles: A review. *Autonomous Intelligent Systems*, 1:1–17, 2021.

[356] X. Yu, Z. Qu, and G. Jin. Robust adaptive filters and smoothers for linear systems with heavy-tailed multiplicative/additive noises. *IEEE Transactions on Aerospace and Electronic Systems*, 60(5):6717–6733, 2024.

[357] W. Yue, J. Ren, and W. Bai. An online outlier-robust extended Kalman filter via EM-algorithm for ship maneuvering data. *Measurement*, 250:117104, 2025.

[358] S. Zair, S. Le Hégarat-Mascle, and E. Seignez. Outlier detection in GNSS pseudo-range/Doppler measurements for robust localization. *Sensors*, 16(4):580, 2016.

[359] N. S. Zewge and H. Bang. A distributionally robust fusion framework for autonomous multisensor spacecraft navigation during entry phase of Mars entry, descent, and landing. *Remote Sensing*, 15(4):1139, 2023.

[360] C. Zhang, M. H. Ang, and D. Rus. Robust lidar localization for autonomous driving in rain. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3409–3415. IEEE, 2018.

[361] H. Zhang. Robust statistics vs. machine learning vs. Bayesian inference: Insights into handling faulty gnss measurements in field robotics. *arXiv preprint arXiv:2504.06015*, 2025.

[362] L. Zhang, G. Pantazis, S. Han, and S. Grammatico. An efficient risk-aware branch MPC for automated driving that is robust to uncertain vehicle behaviors. In *2024 IEEE 63rd Conference on Decision and Control (CDC)*, pages 8207–8212. IEEE, 2024.

[363] T. Zhang and C. Tomasi. Fast, robust, and consistent camera motion estimation. In *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, volume 1, pages 164–170. IEEE, 1999.

[364] W. Zhang and S. Song. Distributionally robust state estimation for highly maneuvering target tracking with model uncertainty and impulsive measurement outliers. *IEEE Sensors Journal*, 2025.

[365] X. Zhang, Y. Chen, X. Zhu, and W. Sun. Robust policy gradient against strong data corruption. In *International Conference on Machine Learning*, pages 12391–12401. PMLR, 2021.

[366] X. Zhang, Y. Chen, X. Zhu, and W. Sun. Corruption-robust offline reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pages 5757–5773. PMLR, 2022.

[367] Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. *International journal of computer vision*, 27:161–195, 1998.

[368] Z. Zhang, X. Zhou, C. Wang, W. Zhao, G. Wu, T. Jiang, and M. Wang. Adaptive robust federal Kalman filter for multisensor fusion positioning systems of intelligent vehicles. *IEEE Sensors Journal*, 24(10):17269–17281, 2024.

[369] J. Zhao and L. Mili. Robust unscented Kalman filter for power system dynamic state estimation with unknown noise statistics. *IEEE Transactions on Smart Grid*, 10(2):1215–1224, 2017.

[370] J. Zhao, G. Yu, and Y. Liu. Assessing robustness of classification using angular breakdown point. *Annals of statistics*, 46(6B):3362, 2018.

[371] P. Zhao and Z. Wan. Robust nonparametric regression under poisoning attack. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17007–17015, 2024.

[372] X. Zhao, Q. Li, C. Wang, H. Dou, and B. Liu. Robust depth-aided visual-inertial-wheel odometry for mobile robots. *IEEE Transactions on Industrial Electronics*, 2023.

[373] X. Zhao, C. Wen, S. M. Prakhya, H. Yin, R. Zhou, Y. Sun, J. Xu, H. Bai, and Y. Wang. Multi-modal features and accurate place recognition with robust optimization for Lidar-visual-inertial SLAM. *IEEE Transactions on Instrumentation and Measurement*, 2024.

[374] Y. Zhe and Z. Hongbo. A novel Bayesian-based INS/GNSS integrated positioning method with both adaptability and robustness in urban environments. *Chinese Journal of Aeronautics*, 37(6):205–218, 2024.

[375] L. Zhllin, Z. Xu, M. Cen, and X. Ding. Robust surface matching for automated detection of local deformations using least-median-of-squares estimator. *Photogrammetric Engineering & Remote Sensing*, 67(11):1283–1292, 2001.

[376] P. Zhou, X. Guo, X. Pei, and C. Chen. T-LOAM: Truncated least squares LiDAR-only odometry and mapping in real time. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13, 2021.

[377] X. Zhou, Z. Wang, and J. Wang. Popov-H_∞ robust path-tracking control of autonomous ground vehicles with consideration of sector-bounded kinematic nonlinearity. *Journal of Dynamic Systems, Measurement, and Control*, 143(11):111004, 2021.

[378] Y. Zhou, Z. Zheng, J. Huang, C. Wang, G. Xu, Y. Xuchen, and B. Zha. Distributed maximum correntropy cubature information filtering for tracking unmanned aerial vehicle. *IEEE Sensors Journal*, 23(9):9925–9935, 2023.

[379] B. Zhu, L. Wang, Q. Pang, S. Wang, J. Jiao, D. Song, and M. I. Jordan. Byzantine-robust federated learning with optimal statistical rates. In *International Conference on Artificial Intelligence and Statistics*, pages 3151–3178. PMLR, 2023.

[380] F. Zhu, Y. Huang, C. Xue, L. Mihaylova, and J. Chambers. A sliding window variational outlier-robust Kalman filter based on Student's t-noise modeling. *IEEE Transactions on Aerospace and Electronic Systems*, 58(5):4835–4849, 2022.

[381] J. Zubizarreta, I. Aguinaga, and J. M. M. Montiel. Direct sparse mapping. *IEEE Transactions on Robotics*, 36(4):1363–1370, 2020.

[382] D. Zügner. *Adversarial Robustness of Graph Neural Networks*. PhD thesis, Technische Universität München, 2022.

TINO WERNER received the B.Sc. and M.Sc. degrees in mathematics from Carl von Ossietzky Universität Oldenburg in 2014 and 2016, respectively, and the Dr. rer. nat. degree in statistics from Carl von Ossietzky Universität Oldenburg in 2020.

Since 2016, he works at Carl von Ossietzky Universität Oldenburg. From 2020 to 2021, he worked at the research institute OFFIS, Oldenburg. Since 2022, he works at German Aerospace Center, Institute of Systems Engineering for Future Mobility, Oldenburg. His research interests include Robust Statistics, Machine Learning and model validation.