Systematic Error Source Analysis of a Real-World Multi-Camera Traffic Surveillance System

Leah Strand
Chair of Robotics, Artificial Intelligence and Real-time Systems
Technical University of Munich
Garching, Germany
leah.strand@tum.de

Jens Honer
Driving Assistance Research (DAR)
Valeo Schalter und Sensoren GmbH
Bietigheim-Bissingen, Germany
jens.honer@valeo.com

Alois Knoll
Chair of Robotics, Artificial Intelligence and Real-time Systems
Technical University of Munich
Garching, Germany
knoll@in.tum.de

***Abstract*—In this paper, we assess the performance of our real-world multi-camera traffic surveillance system along a segment of the A9 Autobahn north of Munich. Its principal component is a Labeled Multi-Bernoulli based tracking module that sequentially fuses the detection data from parallel camera processing pipelines. We present a systematic investigation of the system's characteristic failure modes that lead to a degradation of its performance. To this end, we assess state of the art metrics and performance measures in regard to their suitability for flagging unwanted behavior or failures in real-world multi-object tracking systems. Our analysis is structured into three levels of abstraction: target-level, time-step-level, and track-level. These abstraction levels allow us to systematically approach the analysis from different perspectives and to direct the focus on recurring errors and systemic deficiencies. In particular, the track-level analysis proved to be the most expedient approach since it drew our attention to system challenges like occlusions and other time-correlated detection errors. It further identified the system bias introduced by the adoption of class-dependent object extents. Our analysis is intended to guide the future development effort of our system and to serve as a basis for investigations and improvements of similar systems.**

***Index Terms*—Traffic surveillance system, error source analysis, multi-target tracking, metrics**

I. Introduction

Precise knowledge of the current traffic situation including the location of all traffic participants is a key element to establish more automated transportation systems. Infrastructure-mounted sensors provide a bird's eye view of the situation that goes beyond the field of view of on-board sensors available to individual traffic participants. Making this global view available to all traffic participants allows individuals to make conscious decisions about their own actions with respect to others and thus, improves the overall safety and efficiency of the transportation system.

The *Providentia++* research project and its predecessor *Providentia* [1] established a test field for autonomous driving that features such a smart traffic surveillance system. Its ultimate goal is to provide a real-time and high-precision digital twin of the traffic to facilitate diverse applications. To this end, parts of the test field are equipped with a multi-camera fusion system that supplies real-time information about the location of the traffic participants. The safe operation as well as a focused development effort require a good understanding of the systems characteristics, limits, and failure modes. A plurality of works on performance metrics for tracking systems was published in recent years that cover applications such as the ranking of algorithms [2], [3], metric-driven sensor registration [4] and optimal estimation [5]. On the other hand, large-scale traffic surveillance systems are extensively investigated [6]. However, to the best of our knowledge, detailed error source analyses on traffic surveillance systems were not conducted so far. The main objective of this work is to present a systematic investigation of the system's deficiencies and biases.

For this purpose, we employ several metrics and performance measures that have been proposed for multi-object tracking systems and evaluate their suitability to flag unwanted system behavior or failures: The optimal sub-pattern assignment metric (OSPA) [7] combines localization and cardinality deviations into a unified error per time step. It sparked subsequent metric definitions, including the generalized OSPA (GOSPA) [8] which further allows dividing the cardinality error into missed and false targets. The importance of error type differentiability for the practical usability of metrics and a list of the basic types of tracking errors are highlighted in [9]. The popular CLEAR multiple object tracking (MOT) statistics [10] were developed for the benchmarking of trackers and condense different performance values into a small number of easily comparable values. However, we believe that they are more suitable for comparative analyses and are thus, not included in our error source analysis. Our analysis is structured into three levels of abstraction: target-level, time-step-level, and track-level. These abstraction levels entail different strategies to systematically approach the analysis and to focus the evaluation effort.

The paper is organized as follows. First, we concisely describe the design of the implemented surveillance system in Section II. Second, we present the evaluation methods and measures in Section III and use them to systematically analyze the system in Section IV. Finally, we summarize our results and provide an outlook on future research in Section V.

This work was funded by the German Federal Ministry of Transport and Digital Infrastructure as part of the research project Providentia++.

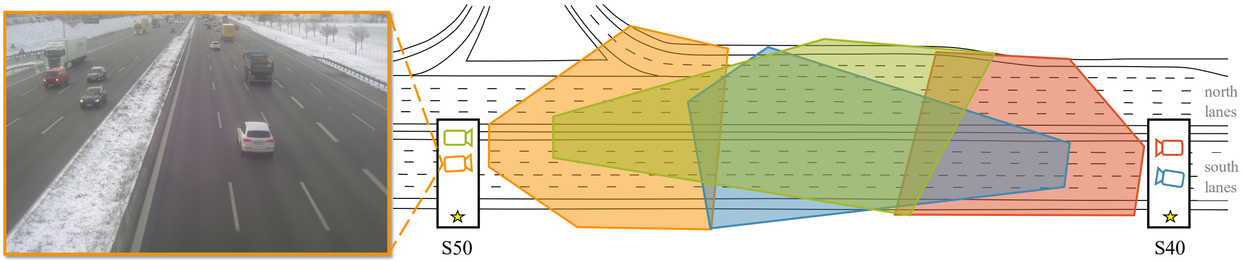


Fig. 1: Test field overview. The outer gantry piers indicated by stars are located at N48°14′29.796″ E11°38′21.083″ (S50) and N48°14′16.170″ E11°38′14.251″ (S40). The field of views of the cameras are colored. An exemplary camera frame is shown.

II. System Setup

In this section, we concisely present the setup of our system. A $450\,\mathrm{m}$ long segment of the A9 Autobahn north of Munich constitutes our region of interest (ROI). It is observed by four cameras that are mounted on two separate gantries, denoted by S40 and S50. An overview of the test field is visualized in Fig. 1. The fundamental system design and its processing blocks are depicted in Fig. 2. First, an object detector condenses the image to its relevant content in parallel pipelines allocated to each camera. The resulting detection lists are then sequentially processed by the tracking module, which fuses the data from all sensors and tracks all traffic objects currently located within the ROI. Details of these steps are outlined in the following.

A. Camera pipeline

Each camera captures raw images from the road scene which are subsequently processed by the detector.

1) Detector: We employ a YOLOv4 network pretrained on the MS COCO dataset [11] for detecting the vehicles in the images. The network output consists of a list of object detections, each comprises a bounding box in normalized image coordinates given by $(\mathit{top}, \mathit{bottom}, \mathit{left}, \mathit{right}) \in [0,1]^4$ as the absolute position and dimension of the axis-aligned bounding box within the image. It further contains an object classification $o \in \{\text{car}, \text{truck}, \text{bus}, \text{bike}, \text{pedestrian}\}$ and the corresponding confidence $c \in [0,1]$ for each detection.

2) Detection postprocessing: An accurate calibration of the sensors is paramount for the precise positioning of the objects and seamless fusion between the different sensors. Our frame of reference is a dedicated HD map that allows us to determine the exact pose of each camera. The resulting transformations are then used to compute the objects' location in Cartesian world coordinates. In order to estimate the relative orientation of the vehicle within each axis-aligned bounding box, we take advantage of the known camera perspective. The lower edge of the bounding box is assumed to be aligned with the front or the back of the vehicle and we choose the lower right $(\mathit{bottom}, \mathit{right})$ or left $(\mathit{bottom}, \mathit{left})$ bounding box corner as reference coordinate r depending on the alignment of the bounding box towards the vanishing point of the image. Then, we project r from the camera-space to the road plane and obtain the Cartesian coordinate (x_r, y_r), which in turn is shifted towards the center of the vehicle by employing a constant class-dependent vehicle width w and length ℓ (see Fig. 3). Finally, we obtain the estimated center coordinate $z = (x_c, y_c)$ on road-level, which, together with the object's classification o and the corresponding classification confidence c, constitute a detection $\tilde{z} = (z, o, c)$. The final output of each camera processing pipeline is a set of n_z detections $\mathbf{z} = \{\tilde{z}_1, \dots, \tilde{z}_{n_z}\}$.

B. Tracking module

The detection data streams are all processed and fused by a single tracking module. The underlying principle is a centralized measurement-to-track fusion that, in principle, offers optimal results since all information is available within the processing node. Furthermore, the input scheduler ensures that the measurements are sequentially processed in the order of their recording.

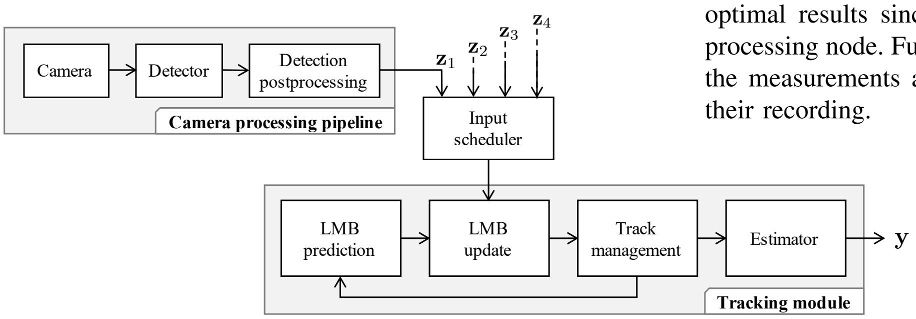


Fig. 2: System setup consisting of one tracking module which is fed by the output of the four parallel camera processing pipelines $\mathbf{z}_1$ to $\mathbf{z}_4$. The final system output is a set of object estimates $\mathbf{y}$.

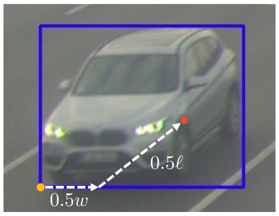


Fig. 3: Estimation of the object center (red) using predefined car extents w, ℓ and the projected reference point (yellow).

1) LMB prediction and update: We use a Labeled Multi-Bernoulli (LMB) tracker [12] to process the detections provided by the cameras. In turn, the multi-object state is represented by an LMB set [13], in which each component is represented by an existence probability, a label, and a multimodal probability density function (PDF) that encodes the knowledge of the object state. We model the PDF with a Gaussian mixture of the kinematic target state vector $\tilde{y} = (x, y, v_x, v_y)$, containing the location x, y and the Cartesian velocity v_x, v_y. The target motion follows a constant velocity motion model [14]. The standard deviation of the acceleration noise longitudinal and lateral to the direction of the road are set to $\sigma_{\mathrm{lo}} = 10\,\mathrm{m\,s^{-2}}$ and $\sigma_{\mathrm{la}} = 1\,\mathrm{m\,s^{-2}}$. The survival probability $P_S(x, y) \in [0, 0.99]$ of the targets is dependent on their location, decreasing towards the exit areas of the ROI. The LMB creates a first-order moment approximation of all possible associations between tracks and detections to calculate the posterior multi-object state and thus, ensures robustness against association errors. In particular, we determine the 10 most likely associations using Murty's algorithm [15]. The detection probability is set to $P_D = 0.95$, the uniform clutter intensity is assumed to be 2×10^{-6} and we employ measurement uncertainties dependent on the camera perspective.

2) Classification processing: The object classification is incorporated by adding a vector $\mathbf{c} \in \mathbb{R}^{|\mathcal{O}|}$, $\|\mathbf{c}\|_1 = 1$ to each track representing the confidence distributed over the classes. The classification from the detection step (o, c) is encoded in a vector $\mathbf{c}_z \in \mathbb{R}^{|\mathcal{O}|}$, where the confidence c is stored at the vector component corresponding to the class o. The remaining components are set to $\frac{1-c}{|\mathcal{O}|-1}$. During the measurement update, the classification is updated using the confidence vector of the last time step $\mathbf{c}_{-1}$ and a damping factor $\gamma = 0.95$: $\mathbf{c} = \mathbf{c}_z + \gamma(\mathbf{c}_{-1} - \mathbf{c}_z)$.

3) Track management: We use a measurement-driven birth model with a position dependent birth probability $P_B \in [0.01, 0.2]$ peaking at the ROI's entry zone. Thus, observations that could not be associated to existing tracks lead to the initialization of new tracks. Furthermore, we employ the standard track cleanup methods of merging, pruning, and capping [12] to ensure real-time capability.

4) Estimator: Tracks that feature an existence probability greater than 0.5 are provided to the output and joined to a set of estimates $\mathbf{y}$. Their kinematic state is estimated by merging the Gaussian mixture of the target state vector and contains the position, Cartesian velocity, and classification as a maximum likelihood estimate.

III. Metric Definitions

In this section, we present the performance values and metrics that are used for the system analysis. First, we introduce general definitions and the matching mechanism. Then, we define the evaluation measures categorized into target-level, time-step-level, and track-level.

A. General definitions

1) Ground truth and estimated data: We assume that all data is collected in a time interval with N_T evaluation steps and the ground truth data $\boldsymbol{x}^{1:N_T}$ is given as a finite sequence of target sets. More precisely, let $I := \{1, \dots, N\}$ be the set of all unique track IDs contained in the whole ground truth dataset. A single target state at time step k with $1 \leq k \leq N_T$ is defined by its track ID $i \in I$, its current position x_i^k and object classification o_i^k, aggregated to the vector $\bar{x}_i^k = (x_i^k, o_i^k)$. Note that in the ground truth data, the object classification is the same for all time steps $o_i = o_i^1 = \cdots = o_i^{N_T}$. Let $\boldsymbol{x}^k$ be the set of n_k targets which exist at time step k, i.e., $\boldsymbol{x}^k = \{\bar{x}^k_{\sigma^k_1}, \dots, \bar{x}^k_{\sigma^k_{n_k}}\}$ where $\{\sigma^k_1, \dots, \sigma^k_{n_k}\} \subseteq I$.
The estimated data $\boldsymbol{y}^{1:N_T}$ is defined similarly with its own track ID set $J := \{1, \dots, M\}$ and the target sets $\boldsymbol{y}^k = \{\bar{y}^k_{\lambda^k_1}, \dots, \bar{y}^k_{\lambda^k_{m_k}}\}$ where $\{\lambda^k_1, \dots, \lambda^k_{m_k}\} \subseteq J$ defines the set of m_k target IDs which exist at time step k.

2) Tracks: Let $\boldsymbol{X} := \{X_1, \dots, X_N\}$ be the set of ground truth tracks. Each track $X_i \in \boldsymbol{X}$ is assigned a finite track state sequence $\tau_i^{1:N_T} = (\tau_i^1, \dots, \tau_i^{N_T})$. For every time step k, the track state is either empty or the current target state if it exists [16]:

$$\tau_i^k = \begin{cases} \{\bar{x}_i^k\} & \bar{x}_i^k \in \boldsymbol{x}^k \qquad \text{(existent state)}, \\ \emptyset & \text{otherwise} \quad \text{(nonexistent state)}. \end{cases}$$

From there, the time of birth ω_i and time of death ν_i of a track X_i are defined as

$$\omega_i = \min_{1 \leq k \leq N_T, \tau_i^k \neq \emptyset} k \quad \text{and} \quad \nu_i = \max_{1 \leq k \leq N_T, \tau_i^k \neq \emptyset} k.$$

Putting everything together, a true track $X_i \in \boldsymbol{X}$ is described by $X_i = (\tau_i^{1:N_T}, \omega_i, \nu_i)$. The same definitions also hold for the set of estimated tracks $\boldsymbol{Y}$.

B. Matching mechanism

In order to quantify the correctness of the system output, the matching between the ground truth data $\boldsymbol{X}$ and the estimated data $\boldsymbol{Y}$ needs to be established. We use the target-level matching mechanism formulated for the OSPA metric [7].
The objective is to obtain an optimal matching between the true target set $\boldsymbol{x}^k$ and the estimated target set $\boldsymbol{y}^k$ for each time step k. Let U^k be the set of all subsets of $\{\sigma^k_1, \dots, \sigma^k_{n_k}\} \times \{\lambda^k_1, \dots, \lambda^k_{m_k}\}$ such that each state from the smaller of both sets is assigned a unique element from the larger set. The optimal matching set $\theta^{k\star}$ is then defined via

$$\theta^{k\star} = \arg\min_{\theta \in U^k} \left(\sum_{(\sigma,\lambda) \in \theta} d^{(c)}(x^k_\sigma, y^k_\lambda)^p \right)^{\frac{1}{p}},$$

where $d^{(c)}(x, y) = \min(d(x, y), c)$ with the Euclidean metric $d(\cdot, \cdot)$ and a cut-off distance parameter $c \in \mathbb{R}$. In this work, we choose $p = 2$ and $c = 7\,\mathrm{m}$. All assignments $(\sigma, \lambda) \in \theta^{k\star}$ with $d^{(c)}(x_\sigma, y_\lambda) = c$ are regarded as unassigned and the assignment set is updated as

$$\theta^{k\star} := \big(\theta^{k\star} \setminus \{(\sigma, \lambda)\}\big) \cup \{(\sigma, 0), (0, \lambda)\}.$$

In addition, all previously unassigned targets that are not already covered in the optimal assignment set $\theta^{k\star}$ are added to it with a 0 partner, i.e.,

$$\theta^{k\star} := \theta^{k\star} \cup \{(\sigma, 0) : (\sigma, \cdot) \notin \theta^{k\star}\} \cup \{(0, \lambda) : (\cdot, \lambda) \notin \theta^{k\star}\}.$$

Using the final matching set $\theta^{k\star}$, each track $X_i \in \boldsymbol{X}$ is assigned a match sequence $\pi_i^{1:N_T} = (\pi_i^1, \dots, \pi_i^{N_T})$ with components

$$\pi_i^k = \begin{cases} -1 & \tau_i^k = \emptyset \quad \text{(nonexistent)}, \\ 0 & (i, 0) \in \theta^{k\star} \quad \text{(no match)}, \\ \lambda & (i, \lambda) \in \theta^{k\star} \quad \text{(match)}. \end{cases}$$

Let $N_{\text{match},i}$ denote the total number of actual matches where the components of $\pi_i^{1:N_T}$ differ from 0 and -1. Similarly, let $N_{\text{exist},i}$ denote the total number of steps where the track exists, i.e, where the components of $\pi_i^{1:N_T}$ are not -1.

Further, let Π_i be the match set containing all the IDs of the estimated states which are matched to the i-th ground truth track over the whole time:

$$\Pi_i = \{\pi_i^k : \pi_i^k \notin \{-1, 0\},\ k \in \{1, \dots, N_T\}\}.$$

The same definitions may be applied in the transposed sense to the estimated tracks $Y_j \in \boldsymbol{Y}$ resulting in the match sequence $\psi_j^{1:N_T}$ and the match set Ψ_j.

Furthermore, let the match start $\tilde{\omega}_i$ and the match end $\tilde{\nu}_i$ of track X_i indicate the time step at which it is firstly and lastly matched:

$$\tilde{\omega}_i = \min_{k, \pi_i^k \notin \{-1,0\}} k \quad \text{and} \quad \tilde{\nu}_i = \max_{k, \pi_i^k \notin \{-1,0\}} k.$$

For estimated tracks Y_j, one can similarly define the match end as

$$\tilde{\nu}_j = \max_{k, \psi_j^k \notin \{-1,0\}} k.$$

C. Target-based evaluation

1) Target categorization: Based on the results from the matching process, a true target x_i^k can be categorized as

$$\text{cat}(x_i^k) = \begin{cases} \text{false negative} & \pi_i^k = -1, \\ \text{matched} & \text{otherwise}, \end{cases} \tag{1}$$

and an estimated target y_j^k can be categorized as

$$\text{cat}(y_j^k) = \begin{cases} \text{false positive} & \psi_j^k = -1, \\ \text{matched} & \text{otherwise}. \end{cases} \tag{2}$$

2) Delayed birth and death: Using the track lifetime bounded by ω_i and ν_i and the match start $\tilde{\omega}_i$ and end $\tilde{\nu}_i$, true targets x_i^k categorized as *false negative* (1) can be further subdivided into

$$\text{cat}'(x_i^k) = \begin{cases} \text{delayed birth} & \omega_i \leq k < \tilde{\omega}_i, \\ \text{delayed death} & \tilde{\nu}_i < k \leq \nu_i, \\ \text{other} & \text{otherwise}. \end{cases} \tag{3}$$

Likewise, estimated targets y_j^k categorized as *false positive* (2) can be further subdivided into

$$\text{cat}'(y_j^k) = \begin{cases} \text{delayed death} & \tilde{\nu}_j < k \leq \nu_j, \\ \text{other} & \text{otherwise}. \end{cases} \tag{4}$$

D. Time-step-based evaluation

1) GOSPA metric: The GOSPA metric [8], a generalization of the widely used OSPA metric [7], quantifies the distance between two sets of targets at each time step. Particularly, it consolidates penalties for localization and cardinality errors between a ground truth reference target set and an estimated target set. As mentioned in [8], the parameter choice of $\alpha = 2$ allows the metric to be decomposed into a separate error for the localization inaccuracy of matched targets, a missed detection error, and a false detection error. In this work, we will therefore use $\alpha = 2$. Localization errors enter into the computation with the Euclidean distance $d(\cdot, \cdot)$ between the matched targets and cardinality errors are counted with the factor $c^p/2$, respectively. Let n_{miss}^k be the number of missed targets and m_{false}^k the number of falsely estimated targets in time step k. With these, the GOSPA distance is computed with

$$d(\boldsymbol{x}^k, \boldsymbol{y}^k) = \left(\sum_{i, \pi_i^k \notin \{-1,0\}} d(x_i^k, y_{\pi_i^k}^k)^p + \frac{c^p}{2} \left(n_{\text{miss}}^k + m_{\text{false}}^k \right) \right)^{\frac{1}{p}} \tag{5}$$

E. Track-based evaluation

1) Track localization accuracy: Given a track X_i and its match sequence $\pi_i^{1:N_T}$, the Root Mean Square Error (RMSE) of the complete trajectory is defined as

$$\text{RMSE}(X_i) = \left(\frac{\sum_{k, \pi_i^k \notin \{-1,0\}} d(x_i^k, y_{\pi_i^k}^k)^2}{N_{\text{match}}} \right)^{\frac{1}{2}}. \tag{6}$$

2) Track categorization: Using the basic types of tracking errors defined in [9], a true track X_i can be categorized into

$$\text{cat}(X_i) = \begin{cases} \text{false negative} & \text{for } |\Pi_i| = 0, \\ \text{fragmented} & \text{for } |\Pi_i| > 1 \\ \text{merged} & \text{for } \Pi_i = \{j\}, |\Psi_j| > 1, \\ \text{unambiguous} & \text{otherwise}, \end{cases} \tag{7}$$

and an estimated track Y_j can be categorized into

$$\text{cat}(Y_j) = \begin{cases} \text{false positive} & \text{for } |\Psi_j| = 0, \\ \text{merged} & \text{for } |\Psi_j| > 1 \\ \text{fragmented} & \text{for } \Psi_j = \{i\}, |\Pi_i| > 1, \\ \text{unambiguous} & \text{otherwise}. \end{cases} \tag{8}$$

Furthermore, we define the track coverage as

$$\text{coverage}(X_i) = \frac{N_{\text{match},i}}{N_{\text{exist},i}}. \tag{9}$$

Let $N_{o,i}$ be the number of actual matches where the object classification o_j^k of the match $\bar{y}_{\pi_i^k}^k = (y_j^k, o_j^k)$ is equal to the true object classification o_i, then the class score is defined as

$$\text{class score}(X_i) = \frac{N_{o,i}}{N_{\text{match},i}}. \tag{10}$$

TABLE I: Total number of true and estimated targets and the partitioning according to their categorization.

	Total	**Matched**	**False negative/missed**	**False positive**
True targets	8735	8070 (92.4 %)	665	-
Estimated targets	8199	8070	-	129

TABLE II: Breakdown of target-level cardinality errors due to track lifetime discrepancies.

	Total	**Delayed birth**	**Delayed death**	**Other**
Missed targets	665	136 (20.5 %)	174 (26.2 %)	355 (53.3 %)
False positive targets	129	-	36 (27.9 %)	93 (72.1 %)

IV. System Evaluation

In this section, the performance of the real-world traffic surveillance system is evaluated using the presented methods. The objective is to estimate the currently achievable positioning accuracy of our system (Section IV-A) and to identify systematic errors that degrade the results (Section IV-B). In the process, we further evaluate the methods on their expediency.

A. Experiment I: Single-object localization accuracy

In the first experiment, a vehicle[1] equipped with a high precision GPS device[2] is used for generating a dataset of ground truth position data to determine the positioning precision of the multi-camera fusion system for this particular vehicle. The dataset consists of 11 passes through the ROI and covers all lanes in both driving directions. Traffic density was low, which results in an ideal object separability. In turn, the results of this experiment can be considered as a baseline for the achievable position under optimal conditions. With the final sequence of matched truth and system positions, the overall track localization accuracy can be computed using (6). The result amounts to an RMSE of $1.50\,\mathrm{m}$. It should be noted that the calculated RMSE value does not reflect the universal positioning accuracy of the system: There is a strong bias since only a single passenger car is considered and the data was captured in low traffic density and ideal weather conditions. Nonetheless, the result yields an estimate of the currently achievable system precision if systematic errors dependent on the object classification and track association failures are reduced to a minimum.

B. Experiment II: Multi-object tracking performance

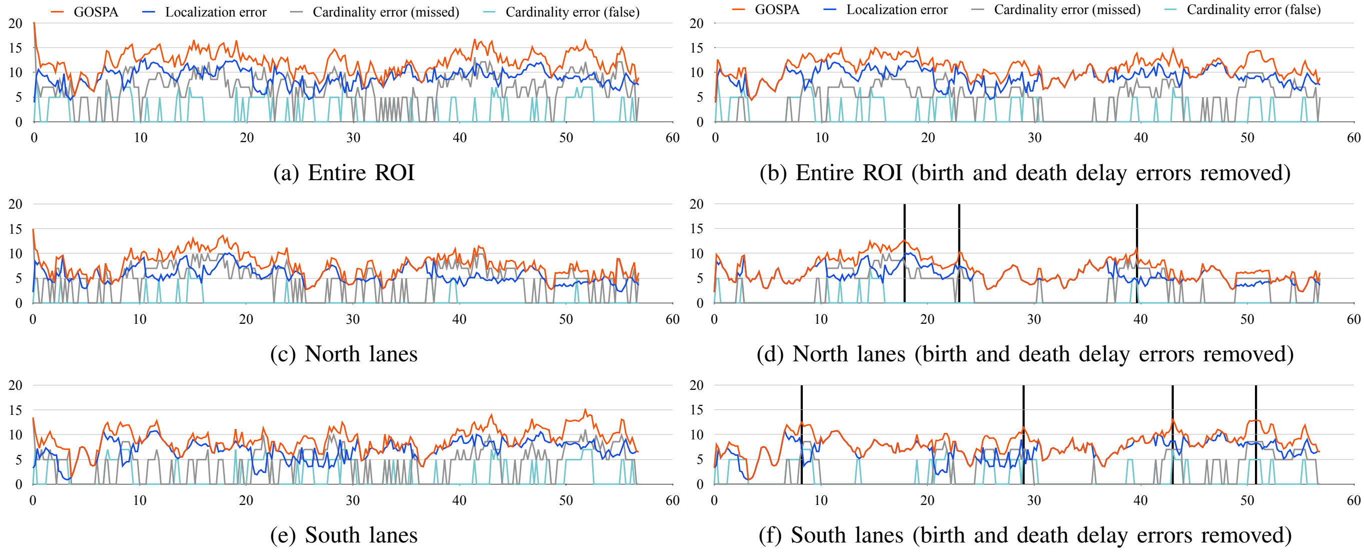
In the main experiment, we focus on the performance during complex multi-object scenarios and analyze potential system biases. Our data basis is a $57\,\mathrm{s}$ sequence which encompasses multiple challenging situations due to high traffic density, many lane changes, and variable object types. We manually labeled all vehicles in the sequence by estimating the true object location using the camera images. The resulting ground truth data comprises 165 tracks, sampled at $5\,\mathrm{Hz}$, i.e., $N_T = 285$ evaluation steps.

1) Target-based evaluation: First, the ground truth data is matched to the output of our tracking system following the target-based matching process described in Section III-B. We start with analyzing the tracking result solely on the base of the target matches. In Table I, all targets are categorized according to the definitions in (1) and (2). For a deeper insight into the false positives and false negatives, we categorize the faults according to the delayed birth and death using (3) and (4). We find that $46.7\,\%$ of all missed targets are due to missing coverage at the start and the end of the true tracks. Similarly, $27.9\,\%$ of all false positives are caused by estimated tracks living on after the death of its corresponding true track. Detailed results are listed in Table II. In conclusion, the target-level analysis helped with characterizing the faulty system behavior due to track lifetime discrepancies and has thus proved to be valuable.

2) Time-step-based evaluation: We calculate the GOSPA metric (5) and its components for all time steps of the annotated sequence; see Fig. 4. First, we pursue the strategy to single out and further analyze steps showing GOSPA peaks in order to identify major systemic deficiencies. We start with the time series GOSPA for the whole scene (Fig. 4a) and recognize that it is difficult to discern steps with particularly striking error characteristics. The issue is that many different and distributed failure modes are combined into the metric which complicates the identification of particular error sources. To mitigate this effect, we follow the approach of restricting the evaluation scope and removing all errors originating from the already identified track lifetime discrepancies. We divide the ROI into two independent evaluation regions, the *northbound* and *southbound* lanes, and calculate separate GOSPA metrics individually (Figs. 4c and 4e). Afterwards, we specifically remove the delayed birth and death errors (Figs. 4b, 4d and 4f). We then manually inspect several selected scenes in which the GOSPA metric exhibits a peak that are highlighted by black bars in Figs. 4d and 4f. It can be noted that in all of these scenes there is discernible erroneous system behavior. Particularly, there are always multiple simultaneous errors, for example three occluded and therefore misdetected vehicles on the northbound lanes at time step $18\,\mathrm{s}$. For our application, it is

[1] BMW 1 Series (F40), dark color

[2] Emlid Reach RS2 using Real-Time Kinematic (RTK) correction provided by the real-time positioning service SAPOS resulting in centimeter-level precision.



(a) Entire ROI

(b) Entire ROI (birth and death delay errors removed)

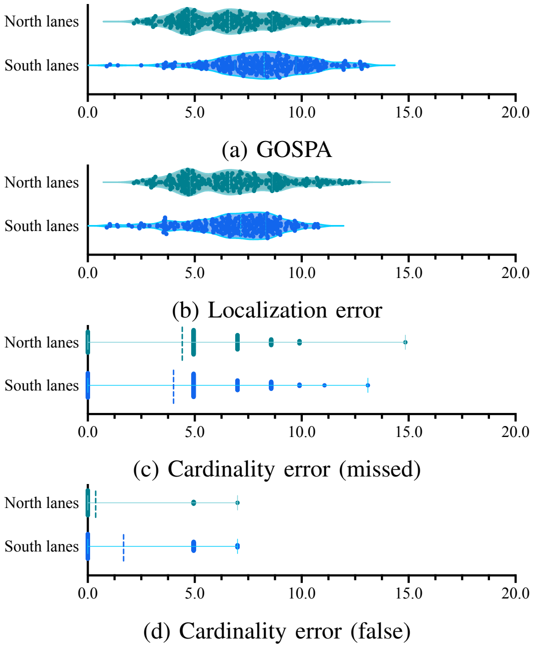
(c) North lanes

(d) North lanes (birth and death delay errors removed)

(e) South lanes

(f) South lanes (birth and death delay errors removed)

Fig. 4: GOSPA metric and its components [m] plotted over time [s] for the entire ROI and separated by *north* and *south* lanes. In the left column, birth and death delay errors are included and in the right column they are excluded. The black bars indicate time steps of manually inspected scenes.



(a) GOSPA

(b) Localization error

(c) Cardinality error (missed)

(d) Cardinality error (false)

Fig. 5: Distribution of the GOSPA metric and its components [m] for the *north* and *south* lanes. Birth and death delay errors are excluded. In Figs. 5c and 5d, the dashed line indicates the mean value.

not ideal to focus the analysis on steps with superimposed errors since severe errors might stay undetected when occurring alone. In conclusion, the approach to analyze particular time steps with large GOSPA values has not been the right way to systematically identify the system's characteristic failure modes. In our analysis, the GOSPA peaks originated from the combination of multiple simultaneous errors and not individual severe errors. The applied restrictions on the evaluation scope were not sufficient to mitigate this effect.

In another approach, we discard the temporal ordering of the GOSPA metric and instead analyze the general distribution of the metric's values. The results from the analysis based on the values of Figs. 4d and 4f are shown in Fig. 5. Here, we can clearly observe that the GOSPA distances restricted to the area of the south lanes concentrate at higher values than those restricted to the north lanes. The same shift is visible in the localization error. In addition, we see that targets are more often missed on the north lanes and that false targets occur more frequently on the south lanes. These are valuable assessments indicating that the system is biased on the location of the vehicles, particularly if they are on the lanes directed to the north or the south. We argue that analyzing the distribution characteristics of the GOSPA metric provides more insight into system characteristics and biases than analyzing the time series.

3) *Track-based evaluation:* Finally, the objective of the track-level evaluation is to focus the analysis on tracks displaying distinctively bad results in order to identify reoccurring error patterns. To this end, we employ the track-based error categorization (7) and (8) to identify particularly critical tracks. Further, we calculate the relative track coverage (9) of all ground truth tracks. We find that $87.9\,\%$ of all ground truth tracks are unambiguously matched to a single estimated track and their mean track coverage amounts to $94\,\%$. On the other hand, the remaining faulty tracks expectably exhibit considerably lower track coverages. Detailed results are listed in Table III. We expect a significant share of the *false negative* target errors to be constituted by these faulty tracks showing lower coverage. On the other hand, it is sensible to assume that the *false*

TABLE III: Total number of true and estimated tracks, the partitioning according to their track-based error categorization and the relative coverage of the true tracks.

		Total	Unambiguous match	Fragmented	Merged	False negative	False positive
True tracks		165	145 (87.9 %)	17	2	1	-
Estimated tracks		184	145	25	7	-	7
True track coverage	(mean)	0.91	0.94	0.71	0.89	0.00	-
	(median)	0.98	0.98	0.69	0.89	0.00	-

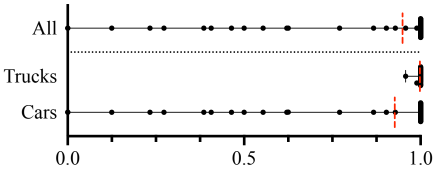
TABLE IV: Breakdown of faulty tracks based on their main source of error determined by manual inspection.

	Total	Match error	Occlusion	Time-correlated detection errors	Inter-camera association
Fragmented true tracks	17	4	12	1	0
Merged estimated tracks	7	4	3	0	0
False positive tracks	7	0	2	3	2

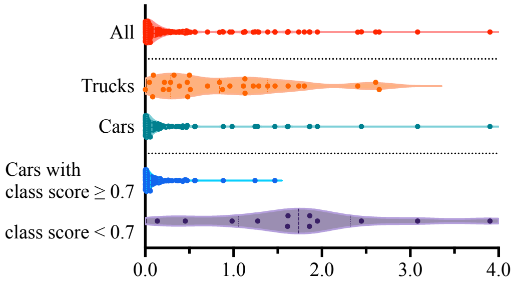
positive tracks contain a major part of all *false positive* target errors. Thus, we conduct a manual data inspection while focusing on the these erroneous tracks which helped to recognize several systematic error patterns. We conclude that in total 8 tracks were wrongly categorized as *fragmented* or *merged* when larger localization errors caused mismatches between the true and estimated tracks. For 17 of the manually inspected tracks, the bad tracking result can be pinpointed to completely or partly occluded objects. The observed occlusions cause not only the fragmentation of 12 tracks, but also 3 track mergers. On top of this, truncated bounding boxes due to only partially visible objects result in 2 *false positive* tracks. This can be explained since the calculation of the object position on street-level fails in these situations. Furthermore, we observe that other detection errors which persist over multiple time steps cause in total 4 faulty tracks. For two *false positive* tracks, the inter-camera measurement association fails which causes the duplication of the tracks, most probably due to remaining calibration inaccuracies. The one *false negative* ground truth track is uncovered because it is born directly before the sequence's end. The results from this manual inspection are summed up in Table IV.

Our method for computing the objects' center positions as presented in Section II-A2 is strongly influenced by the classification of the objects since it employs class-dependent constant object extents. Thus, we measure the influence of the object classification on the localization accuracy quantified with the track-based RMSE (6). In this analysis, we only consider the 145 unambiguously matched ground-truth tracks in order to minimize the influence of track association inconsistencies on the localization error. First, we partition the tracks based on their true classification and notice that the track RMSE of cars concentrate at values below 0.5 m, while the track RMSE of trucks are generally spread out below 2.0 m (see Fig. 6b). This is plausible since the larger extent of trucks also imply potentially higher localization errors when computing the center position. On the other hand, multiple car tracks display an unstable object classification signified by a low class score (see Fig. 6a). After dividing the RMSE analysis of the car tracks based on high and low class scores, we determine that nearly all of the low-accuracy car tracks are simultaneously often misclassified. This effect can be explained since the misclassification of an object causes the inaccurate computation of its center position due to employing unsuitable object extents.

The conducted track-based analysis helped in directing the focus on major error sources. For this purpose, the straightforward categorization of tracks using their match sequence proved to be very effective in highlighting critical tracks. In turn, the further analysis of these tracks drew our attention to several systemic deficiencies. Moreover, it helped to determine that the localization accuracy of a track is biased on its object classification, while especially misclassifications have a deteriorating effect.



(a) Class score



(b) RMSE

Fig. 6: Class score (mean in red) and distribution of the RMSE for all unambiguously matched ground truth tracks and partitioned based on the object classification. The RMSE of car tracks are further divided into high and low class scores.

V. Conclusion and Outlook

In this work, we analyzed the performance of our real-world multi-camera surveillance system. We presented an estimate on the currently achievable positioning accuracy and identified failure modes and biases of the system. To this end, we conducted analyses on target, time step, and track-level and evaluated the strategies on their suitability to a systematic error source analysis. We found that a deeper analysis of the target-level errors is valuable to understand the characteristics of the system such as track lifetime discrepancies. Furthermore, the strategy to focus the analysis on time steps which show a peak in the GOSPA metric turned out not to be the right approach since it highlighted only simultaneous errors and no solitary severe errors. This shortcoming might be leveraged if the scope of the evaluation is further scaled down by conditioning the GOSPA metric to situations, targets, or locations one suspects to contain notable error characteristics. In addition, we determined a distinct bias dependent on the location of the objects by analyzing the distribution characteristics of the GOSPA metric restricted to subareas of our ROI. We found the distribution analysis more informative than the time series GOSPA evaluation.

Analyzing the error sources on track-level was the most expedient approach to direct the focus on reoccurring failure patterns. Further, we determined that the localization accuracy of the tracks is biased on the object classification. However, in future analyses, the amount of evaluation data should be extended to further enhance the validity of the results. The error source analysis may be improved by using a track-based matching that incorporates the track history or penalizes switches [16], [17]. Another approach would be to implement the matching by maximizing the detection and association success [18]. This might reduce wrongly *fragmented* or *merged* flagged tracks.

The analysis we conducted serves to improve the tracking system deployed in Providentia++. In particular, we systematically identified the need for a more robust occlusion handling as well as a handling of strongly correlated errors and calibration inaccuracies. Our presented approach is general so that other perception and tracking systems can benefit from similar analyses.

Acknowledgment

We would like to thank Venkatnarayanan Lakshminarasimhan, Christian Creß, and Walter Zimmer for the collective work on the surveillance system, in particular for the construction of the hardware setup. We further want to acknowledge Venkatnarayanan Lakshminarasimhan and Xinyi Li for the recording of the ground-truth GPS data. Furthermore, we want to thank all project partners for the cooperation, especially 3D Mapping Solutions for providing the HD map. Finally, we would like to thank the Bavarian State Office for Survey and Geoinformation for the access to the real-time positioning service SAPOS.

References

[1] A. Krämmer, C. Schöller, D. Gulati, V. Lakshminarasimhan, F. Kurz, D. Rosenbaum, C. Lenz, and A. Knoll, "Providentia - a large-scale sensor system for the assistance of autonomous vehicles and its evaluation," *arXiv preprint arXiv:1906.06789*, 2019.

[2] Y. Xia, K. Granström, L. Svensson, and Á. F. Garca-Fernández, "Performance evaluation of multi-Bernoulli conjugate priors for multi-target filtering," in *20th International Conference on Information Fusion*, 2017, pp. 1–8.

[3] J. Smith, F. Particke, M. Hiller, and J. Thielecke, "Systematic analysis of the PMBM, PHD, JPDA and GNN multi-target tracking filters," in *22th International Conference on Information Fusion*, 2019, pp. 1–8.

[4] Á. F. García-Fernández, M. L. Hernandez, and S. Maskell, "An analysis on metric-driven multi-target sensor management: GOSPA versus OSPA," in *IEEE 24th International Conference on Information Fusion*, 2021, pp. 1–8.

[5] Á. F. Garca-Fernández and L. Svensson, "Spooky effect in optimal OSPA estimation and how GOSPA solves it," in *22th International Conference on Information Fusion*, 2019, pp. 1–8.

[6] C. Creß and A. Knoll, "Intelligent transportation systems using external infrastructure: A literature survey," *arXiv preprint arXiv:2112.05615*, 2021.

[7] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, 2008.

[8] A. S. Rahmathullah, Á. F. Garca-Fernández, and L. Svensson, "Generalized optimal sub-pattern assignment metric," in *20th International Conference on Information Fusion*, 2017, pp. 1–8.

[9] I. Leichter and E. Krupka, "Monotonicity and error type differentiability in performance measures for target detection and tracking in video," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2003–2009.

[10] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *EURASIP Journal on Image and Video Processing*, vol. 2008, no. 1, p. 246309, 2008.

[11] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[12] S. Reuter, B.-T. Vo, B.-N. Vo, and K. Dietmayer, "The labeled multi-Bernoulli filter," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3246–3260, 2014.

[13] B.-T. Vo and B.-N. Vo, "Labeled random finite sets and multi-object conjugate priors," *IEEE Transactions on Signal Processing*, vol. 61, no. 13, pp. 3460–3475, 2013.

[14] Y. Bar-Shalom, X. Li, and T. Kirubarajan, *Estimation for Kinematic Models.* John Wiley & Sons, Ltd, 2002, ch. 6, pp. 267–299.

[15] K. G. Murty, "An algorithm for ranking all the assignments in order of increasing cost," *Operations Research*, vol. 16, no. 3, pp. 682–687, 1968.

[16] Á. F. Garca-Fernández, A. S. Rahmathullah, and L. Svensson, "A metric on the space of finite sets of trajectories for evaluation of multi-target tracking algorithms," *IEEE Transactions on Signal Processing*, vol. 68, pp. 3917–3928, 2020.

[17] M. Beard, B. T. Vo, and B.-N. Vo, "OSPA(2): Using the OSPA metric to evaluate multi-target tracking performance," in *International Conference on Control, Automation and Information Sciences (ICCAIS)*, 2017, pp. 86–91.

[18] J. Luiten, A. Osep, P. Dendorfer, P. Torr, A. Geiger, L. Leal-Taixé, and B. Leibe, "HOTA: A Higher Order Metric for Evaluating Multi-object Tracking," *International Journal of Computer Vision*, vol. 129, no. 2, pp. 548–578, 2021.