Semi-autonomous reference data generation for perception performance evaluation

Thomas Tatschke, Franz-Josef Färber, Erich Fuchs
FORWISS
University of Passau
Passau, Germany
Email: {tatschke, faerber, fuchse}@forwiss.uni-passau.de

Leonhard Walchshäusl, Rudi Lindl
Vehicle Sensors and Perception Systems
BMW Group Research and Technology
Munich, Germany
Email: {leonhard.walchshaeusl, rudi.lindl}@bmw.de

***Abstract*—In the development phase of perception systems (e. g. for advanced driver assistance systems) general interest is pointing towards the performance of the respective detection and tracking algorithms. One common way to evaluate such systems relies on simulated data which is used as a reference. We present a semi-autonomous method, which allows the extraction of reference data from sensor recordings (including data at least from a camera and a distance measuring sensor device). Furthermore, we show how to combine these reference data with the output from the detection or fusion system and how to derive performance statistics of the system. As the generated reference information can be stored along with the sensor recordings, this method also facilitates the comparison of different software versions or algorithm parameters.**

Keywords: Performance evaluation, reference data generation.

I. Introduction

Machine-based object recognition and tracking is applied in numerous applications like video surveillance, robotics or advanced driver assistance systems. Especially perception systems which allow for automotive active safety systems benefit from a performance evaluation, since incapable false alarm and detection rates would represent a major setback to the driver acceptance or could even endanger other road users. In addition, a continuous performance evaluation during the development phase of a perception system helps to discover the influence of system parameters or algorithmic modifications on the detection or tracking performance.

Perception systems are usually evaluated against a ground truth which can be acquired by dedicated sensor data, simulation or pre-recorded data. In the former case, a dedicated sensor is utilized to gather additional sensor information, such as e. g. radio signals, to determine relevant object positions. This approach can be rather costly and might introduce further problems, like shadowing effects or poor synchronization.

Simulated ground truth requires an emulation of the scenery (reference objects, background, object dynamics, etc.) and of the sensing principles. On the one hand this approach can provide an almost unlimited amount of testing data, on the other hand setting up or extending a simulation model might be rather time-consuming and complicated, particularly for complex scenes or within multi-sensor systems. In general, a simulation approach can only offer an approximation of the sensing principles and a given scenery.

A ground truth database can also be constructed by a human operator who manually annotates the pre-recorded sensor data. In addition to the performance evaluation the annotated real-world data can be utilized as sample data in the training process of a classifier.

In order to achieve meaningful evaluation results the ground truth database should reflect various varieties of background, illumination and object appearances. Moreover, the dynamic behavior has to be considered if the tracking is evaluated. Unfortunately, all these constraints lead to either complex simulation models or huge amounts of recorded data which has to be annotated frame by frame.

In conclusion, setting up a ground truth database is a laborious task. Our semi-automatic reference generation technique tries to reduce the effort with a trade-off between complex simulation and pure manual annotation.

A. Related Work

Several performance evaluation approaches for image-based object detection have been proposed in the literature.

Bertozzi et al. [1] describe a system for evaluating image-based pedestrian detection algorithms. They introduce a tool which allows supervised video sequence annotations, sequence annotations by the algorithm being tested, annotation matching and analysis. A human operator who examines the pre-recorded sequences frame by frame achieves about 100 frames per hour.

Dörmann [2] and Jaynes [3] present performance evaluation systems (ViPER and ODViS) which provide interfaces for ground truth generation, metrics for evaluation and tools for visualization of video analysis results.

Vogel [4] utilizes a real-time-kinematic differential GPS system to locate and track pedestrians with centimeter accuracy. The data from this dedicated sensor are used to evaluate a multi-sensor perception system.

In the application field of video surveillance Black et al. [5] present an automatic generation of ground truth for large datasets of video with a fixed mounted camera. As an alternative to manual ground truthing, isolated object tracks are used to construct a comprehensive set of pseudo-synthetic video sequences. Nevertheless, a sufficient useful generation of pseudo-synthetic ground truth data from a moving camera or even from multiple sensors is a very difficult task.

The contribution of this paper is a semi-automatic annotation of object tracks in three dimensions facilitating high labeling speed and both classifier and tracker performance analysis. The accuracy versus labeling speed ratio is arbitrarily eligible by means of a new divide and conquer annotation strategy.

B. Overview

The underlying paper is structured in two main chapters: The first one addresses the generation of reference data. It starts with the presentation of the basic concept (cf. section II-A) and explains the extraction of reference information from recorded data (cf. sections II-B and II-C). In order to support the extraction process a flexible key-frame methodology is introduced in section II-D. This chapter finishes with an introduction in the graphical user interface (cf. section II-E) and the storage of reference information in XML files (cf. section II-F). The chapter III on perception performance evaluation shows how to use the obtained reference data to assess the respective perception system by providing statistical measures. Before being able to calculate performance rates in section III-C the reference data has to be associated to the results of the perception system (cf. section III-A) and limited to the measuring space of the sensor devices (cf. section III-B). The chapters IV and V are summarizing the paper and are giving prospects to future work.

II. Reference data extraction

The following remarks are based on the assumption that the perception system to be evaluated can be run off-line on previously recorded sensor data and performs identically to its online (e. g. in a demonstrator vehicle) operation. For the presented reference data extraction the sensor data recordings have to include at least a camera and a distance measuring sensor device. We are considering a laser scanner as an example in the following.

First of all it is necessary to distinguish between representations of real existing objects to be detected in a recorded scenario, so-called *reference objects*, and objects generated by the perception system (*perception objects*). In our application at hand reference objects represent traffic participants, e. g. vehicles, pedestrians, etc., which should be detected and tracked by the perception system. These reference objects provide a quite objective overview of the scenario whereas the perception objects form the subjective view of the underlying perception system. If both views coincide, the respective perception system has interpreted the scene perfectly.

Whereas the perception objects are the output of a detection and tracking system, the information of the reference objects has to be extracted with user interaction. This first phase of the performance evaluation is called reference data extraction and has to be performed only once per recorded scenario.

A. Concept

The process of generating reference information basically consists of an assisted labeling of interesting objects in the sensor data. In doing so, reference objects in a single frame are represented as three dimensional cuboids with the parameter position $(x, y, z) \in \mathbb{R}^3$, width w, height h, depth d and orientation given by the quaternion $(\phi, a_x, a_y, a_z) \in (2\pi, 2\pi) \times \mathbb{R}^3$. Thereby the x-axis of the reference object's immanent local coordinate system (cf. figure 1) characterizes per definition the object's moving direction. Additionally further properties like a name, a classification type and other dynamically configurable attributes can be assigned to the reference object. We chose this type of model for our reference data because a cuboid should enclose most relevant real-world objects with a sufficient accuracy.

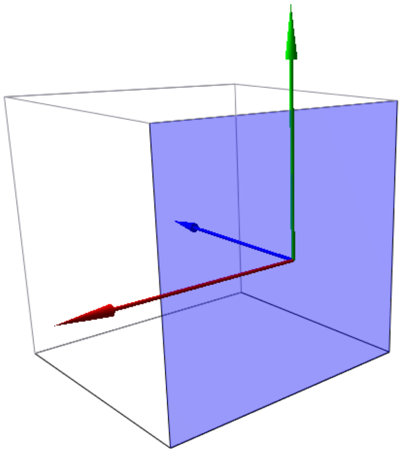


Figure 1. Cubic model of a reference object with orientation vector (blue arrow) and its face towards the imaging sensor (blue face).

B. Extraction of reference objects' dimension and 2D position

The real process of reference generation is twofold. The first step consists of the extraction of the reference objects' dimension and 2D position in a camera frame of the recorded data. This is mainly done by annotating each interesting object which is visible in the camera image as follows:

In the camera frame such kind of cuboid (representing a reference object's model) with the dimensions of the visible object to be extracted can be drawn. This cuboid has to be positioned in a way that it encloses (with its front closest to the camera) the object to be acquired completely in the image. Thereby the orientation of the cuboid's local coordinate system should be aligned with the moving direction of the object itself. Thus the position, dimension and orientation of the object to be captured from the image is fixed at least in 2D (cf. figure 2). By means of the camera's (re-) projection properties the respective magnitudes in 3D space can be calculated from this data if the object's distance from the camera is given. This re-projection step is done automatically by choosing a default value as depth of the respective object at hand (e. g. 3m in front of the camera). In doing so, world coordinates in 3D space are obtained for the reference object.

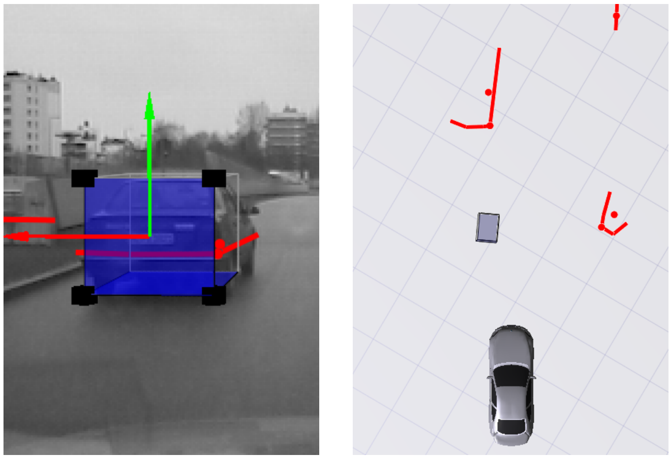


Figure 2. Reference object (blue cuboid) extracted from the camera image only; laser scanner data is shown in red color.

It must be pointed out that the reference information generated in this manner does not reflect the actual dimensions of the real object. Due to the unknown object's depth and the adopted default value for the distance it is generally (as a result of the underlying camera mapping function) a scaled version of the real object (except the special case that the real object to be extracted has the same distance from the camera as the assumed default value). Consequently the dimensions of the object calculated this way equal the ones of the object to be extracted up to a scalar factor.

C. Extraction of reference objects' 3D position

In the second step of generating reference information each reference object is assigned to its effective depth information. In order to determine the accurate distance and the exact dimensions of the object to be extracted, it is essential that the real object is not only seen in the camera's picture but also has induced a measurement in at least one distance measuring sensor device. Based on this distance measurement the reference object generated up to this point is moved along the camera's viewing rays and scaled in such a way that

- the distance measurement is located on the cuboid's face next to the camera (translative component) and that
- the impression of the cuboid in the image is not changed (scaling component), i. e. the cuboid has not changed its position nor orientation in the image.

That is to say the reference object is still enclosing the target object in the image and has the same distance to the camera as the target object has (cf. figure 3). As a result of the just performed scaling of the reference object depending on its depth it now approximately incorporates the dimensions of the target object.

D. Interpolation between frames

The extraction process described above results in a reference object with certain attributes (position, orientation, dimensions, etc.) for a distinct time. In most cases this kind of information is not sufficient to evaluate an algorithm or a system. In fact, data of the reference object's attributes at a series or interval of time within the life-cycle of the object (i. e. the time, in which the object can be perceived in the sensor system) is required. In order to reduce the efforts of extracting these data in every sensor cycle and camera frame, we decided to realize a so-called key-frame concept. - a continuous variation of the objects' position, orientation, etc admitted. Its main idea is that the attributes, like position, orientation (and also dimensions) of the reference object, which has been extracted from two sensor cycles or camera frames $a, b \in F$, $a < b$ (whereas F denotes the set of all frames within a recording) on the basis of a common target object, are interpolated in all frames $c \in F$ with $a < c < b$. Let $p : F \to \mathbb{R}^3$ assign to each frame a position of a fixed reference object, then the position of the reference object in frame c is given by

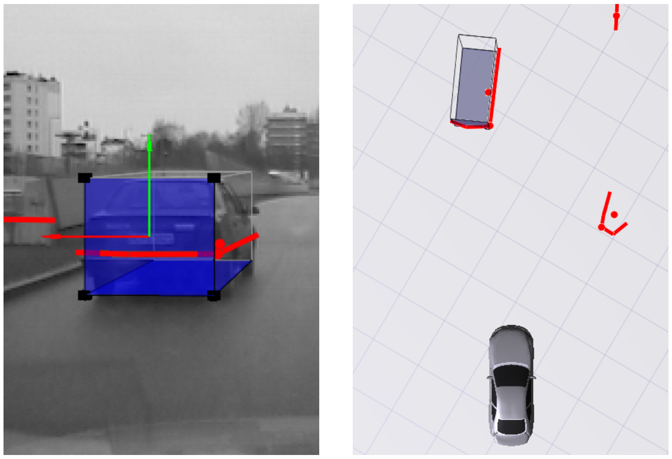


Figure 3. Reference object (blue cuboid) with actual 3D position set from laser scanner measurements (red).

$$\forall_{c \in F, a<c<b} \quad p(c) = p(a) + \frac{p(b) - p(a)}{b - a} \cdot (c - a) \qquad (1)$$

The interpolation of the orientation (coded in quaternions) and of the object's dimensions is calculated in an analogous manner to equation (1). Thus, it is not necessary anymore to extract reference information from every sensor cycle or camera frame. But it is possible to skip frames and to label target objects in every third frame for instance. In our application - the tracking of preceding vehicles - it is even possible to skip a lot more frames as long as the vehicle in front does not perform many manoeuvres. In the case of an insufficient mapping of the object's movements the adding of new key-frames is supported in between two existing ones to improve the modeling of the object's trajectory.

As a result of this a reference object over time is represented as list of position, orientation and other attributes at distinct sensor cycles or camera frames (key-frames). In these key-frames the objects' immanent information has been extracted from the sensor data and describes the properties of the

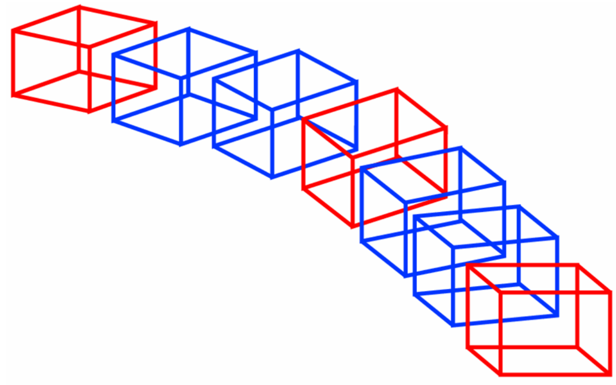
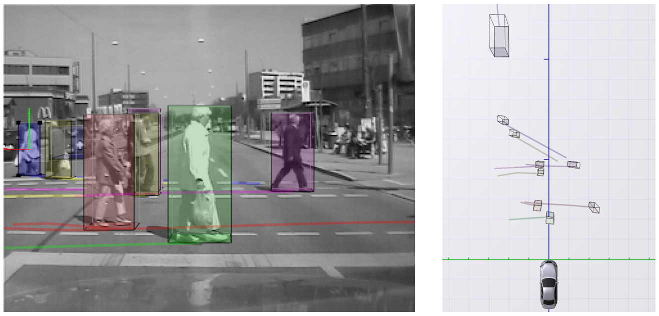


Figure 4. Track of a reference object's path, including key-frames (red) and interpolated data (blue).

real target object. In all frames between two key-frames the reference object's properties are interpolated as specified before. That way the lifetime of a reference object is restricted to the time between its first and its last key-frame.

E. User Interface

The objective of the presented user interface is twofold: Firstly, it has to support multi sensor perception systems and multi dimensional reference data. Secondly, the interface should reduce the average workload for annotating objects and help the user in keeping control even of complex scenes.



(a)

(b)

Figure 5. Snapshot of a complex annotated scene containing 12 pedestrians and one vehicle. (a) Sensor interaction view. (b) Virtual interaction view.

An intrinsic and extrinsic calibration of the multi sensor system is mandatory for a coherent visualization of reference objects and 3D sensor data within a virtual 3D window (cf. figure 5(b)). The user is able to freely navigate and manipulate object attributes like position, dimension and orientation within this virtual environment. This view is mostly used to control distance information or perform fine tuning. The images of visual cameras are displayed in separated windows (cf. figure 5(a)). In order to support the annotation process these video displays are augmented with available 3D reference and sensor data. The video window is mainly used for the instantiation of new reference objects or the supervision of object tracks. Moreover, object parameters like the lateral position, width or height can be modified in this view. The distance attribute is adjusted in two ways: Firstly, the position and dimension within the image plane are fixed. In accordance with the change in distance the real object dimension is modified. Secondly, the real object's dimension is fixed whereas the position and dimension within the image plane is altered. Most of these tasks can be performed both by mouse or keyboard interaction which further enhances the usability. Moreover, different colors, transparency modes and reference objects' trajectories support the user to retain control in complex scenarios (cf. figure 5).

We propose following divide and conquer strategy to speed up the labeling process:

1) The first and last appearance of an object within the recorded data will set up the initial and the final key-frame for that object.
2) If the interpolation between the beginning and the end of the current section does not satisfy the precision demands, a new key-frame is inserted in the middle between the lower and upper bound of the current section. Thereby, two new sections are created.
3) Step 2 is repeated until all sections satisfy the precision demands.

Using the proposed strategy we were able to annotate the complex scene from above with 500 frames including 12 pedestrians and one vehicle in less than ten minutes.

F. Storage of reference data

The resulting reference information, i. e. a list of reference objects together with their attributes and key-frames, can be stored into a file in XML format. This facilitates an easy and human readable way to preserve the results from the reference extraction process. A sample extract from such kind of file is shown below:

```
<AssessmentData>
  <ReferenceObject name="Car_1" type="vehicle">
    <Attributes>
      <parameter name="color" value="silver"/>
      <parameter name="size" value="medium"/>
      [...]
    </Attributes>
    <keyframe timestamp="16189">
      <position x="21.7445" y="-0.2729" z="1.0342"/>
      <direction1 x="0.06451" y="0.8247" z="-0.00259"/>
      <direction2 x="0.03216" y="-0.0009" z="0.7674"/>
    </keyframe>
    <keyframe timestamp="16200">
      <position x="31.3841" y="-3.2368" z="0.8723"/>
      <direction1 x="0.0466" y="0.5841" z="-0.0018"/>
      <direction2 x="0.0379" y="-0.0001" z="0.9060"/>
    </keyframe>
    [...]
  </ReferenceObject>
  <ReferenceObject name="Ped_1" type="pedestrian">
  [...]
  </ReferenceObject>
  [...]
</AssessmentData>
```

Furthermore, it is supported to add, edit or delete further objects and keyframes later on. The application also allows to assign any attribute (in our sample file it is color and size) to the objects. As the positions of the reference objects are stored relative to the sensor system (or ego vehicle in our case) even a change of the sensor system's dynamical model can be compensated in the reference information.

III. Perception performance evaluation

In order to draw conclusions regarding the performance of a perception system, it is essential to assess its output in relation to the real scene. In our application the ground truth information about the real scenario is given by the set of extracted reference objects (cf. section II-B). Thus it has to be assured that the reference data has been generated according to the focus of the underlying perception system. That means if the perception system is specialized in detecting and tracking pedestrians the respective set of reference data should at least include all pedestrians in the scenario and not only vehicles.

A. Association of perception and reference data

To carry out an evaluation of the perception system's output it is necessary to identify both accordances and variations between the set of perception objects P_f and the set of reference objects R_f per camera frame $f \in F$. As our approach assesses only on a binary basis, i. e. it decides if the perception system has or has not detected the target object correctly, and does so far not take into account the precision of the objects' position, orientation and dimension, it is adequate to calculate a one-to-one assignment between the elements of the sets P_f and R_f. This assignment can be modeled as a state-of-the-art data association problem: The objective is to associate every perception object $p \in P_f$ just to one reference object $r \in R_f$. Let $d : P_f \times R_f \to \mathbb{R}$ denote the distance between two objects this problem can be solved with standard data association techniques (e. g. GNN, bipartite matching, etc.). In doing so, the objects p, r will only be assigned if their distance does not exceed a certain pre-defined threshold $d(p, r) < \sigma$ with $\sigma \in \mathbb{R}$.

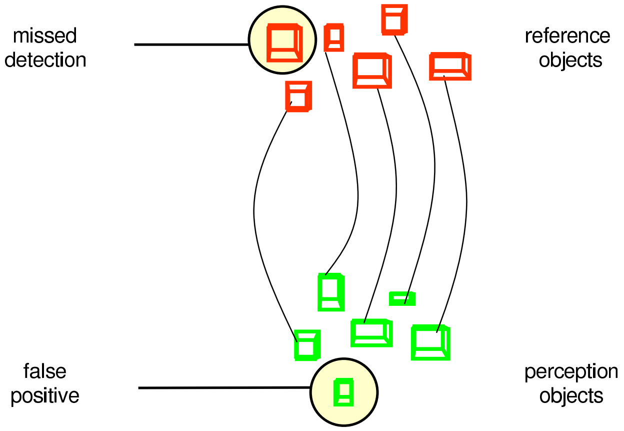


Figure 6. One to one association of perception objects (green) and reference objects (red)

In general, the determination of the assignment $A_f \subset P_f \times R_f$ should not be a problem since no real-time constraints have to be fulfilled for its calculation. Furthermore, one can rely in most cases on the assumption that the objects are not closely spaced and therefore well separated, which makes their association easier. In the case of closely spaced objects, e. g. a group of pedestrians, different association algorithms can be run in parallel or a recourse on interactive support is possible.

B. Performance evaluation space

In order to generate statistical data from the associated object data per frame in the next step, it makes sense to restrict the space $E \subseteq \mathbb{R}^3$, in which assignments and missing assignment are used for calculating performance measures. The restriction aims at selecting these objects from the set of reference data R_f at frame f. They are consulted for the assessment of the perception system's output, thus all from $\{r \in R_f \mid pos(r) \in E\}$, whereas $pos : R_f \to \mathbb{R}^3$ and $pos : P_f \to \mathbb{R}^3$ respectively are denoting the position of an object. In this regard an upper bound for E is given by the measurement range covered by the sensor system, as it is not reasonable to assess every object outside the measuring range, which was included in the reference information but could obviously not be detected by the perception system, as a missing detection for example.

Let S denote the set of sensors and let $FOV(s) \subset \mathbb{R}^3$ be the field of view of the sensor $s \in S$. Then the area, which is covered by at least $k \in \mathbb{N}$ sensors, $k \leq |S|$, is given by

$$E_{S,k} \overset{\text{def}}{=} \bigcup_{I \subseteq S, |I|=k} \bigcap_{s \in I} FOV(s). \qquad (2)$$

In the case $k = 1$ the measuring range that is usually used for performance evaluation of a perception system is obtained.

If the focus is on the evaluation of a subsystem $T \subseteq S$ of the complete sensor system (e. g. object detection and tracking with a camera only), the perception system has to be configured in such a way that only the interesting subsystem delivers results (i. e. perception objects). Furthermore, the measurement range has to be constrained to

$$E_{T,1} = \bigcup_{s \in T} FOV(s) \qquad (3)$$

for instance. This methodology facilitates the performance analysis of all sensor subsets without additional efforts in generating reference information.

Moreover a so-called 'mandatory area'$M \subseteq \mathbb{R}^3$ can be defined which aims at listing the relevant statistical performance measures built upon data from this area separately. This information can be of special interest if there is an application driven critical area with regard to the reliability of the perception system's output (e. g. the 'region of no escape' in front of a moving vehicle equipped with a collision mitigation application, cf. [6]).

C. Calculation of performance measures

On the basis of the object association A_f and the sets P_f and R_f a partition of the objects in the frame f can be calculated as follows:

- If a reference object $r \in R_f$ has been assigned to a perception object $p \in P_f$, i. e. $\exists_{a \in A_f} a = (p, r)$, then the object r has been detected (or tracked) correctly by the perception system. These objects can be summarized in the set
$$HIT_f \overset{\text{def}}{=} \left\{ p \in P_f \;\middle|\; \exists_{a \in A_f} a = (p, r), r \in R_f \right\} \tag{4}$$
of all correct perception results.
- If a reference object $r \in R_f$ is lacking an assignment to a perception object $p \in P_f$, i. e. $\forall_{p \in P_f} \nexists_{a \in A_f} a = (r, p)$, then the perception system has missed the respective target object (missed detection). Furthermore, the reference object should be located at the measuring range of the sensor system, i. e. $pos(r) \in E_{S,k}$. The set of all missed objects can be written as
$$MD_f \overset{\text{def}}{=} \left\{ r \in R_f \;\middle|\; \nexists_{a \in A_f} a = (r, p) \wedge pos(r) \in E_{S,k} \right\}. \tag{5}$$
- If a perception object $p \in P_f$ has not been assigned to any reference object, i. e. $\forall_{r \in R_f} \nexists_{a \in A_f} a = (r, p)$, then the underlying perception system has created a ghost object, a so-called false alarm. The set of all false alarms can be represented by
$$FP_f \overset{\text{def}}{=} \left\{ p \in P_f \;\middle|\; \forall_{r \in R_f} \nexists_{a \in A_f} a = (r, p) \right\}. \tag{6}$$

Furthermore, so-called classification errors can be determined as long as the perception system itself carries out the classification task. Let $t_1 : R_f \to C$ and $t_2 : P_f \to C$ respectively denote the classification type of an object, then the set of all wrong classified objects in frame f can be written as

$$CE_f \overset{\text{def}}{=} \left\{ p \in P_f \;\middle|\; \exists_{r \in R_f} (p, r) \in A_f \wedge t_2(p) \neq t_1(r) \right\}. \tag{7}$$

Over a series of frames F the measures for false alarms (FP), missed detections (MD), the detection rate (HIT) and the classification error (CE) can be determined from these sets as follows:

$$FP \overset{\text{def}}{=} \frac{\sum_{f \in F} |FP_f|}{\sum_{f \in F} |P_f|} \tag{8}$$

$$MD \overset{\text{def}}{=} \frac{\sum_{f \in F} |MD_f|}{\sum_{f \in F} |R_f|} \tag{9}$$

$$HIT \overset{\text{def}}{=} 1 - MD \tag{10}$$

$$CE \overset{\text{def}}{=} \frac{\sum_{f \in F} |CE_f|}{\sum_{f \in F} |A_f|} \tag{11}$$

There is also the possibility to set the failures in relation to the number of frames as e. g. $FP \overset{\text{def}}{=} \sum_{f \in F} \frac{|FP_f|}{|F|}$. Furthermore, it can be useful to calculate the preceding measures not only for the whole set of objects, but for selected types of objects separately (e. g. the hit rate of pedestrians), or to restrict the measures to a certain area of interest (cf. subsection III-B), e. g.

$$FP_{|M} = \frac{\sum_{f \in F} |\{p \in FP_f \mid pos(p) \in M\}|}{\sum_{f \in F} |\{p \in P_f \mid pos(p) \in M\}|}. \tag{12}$$

IV. Conclusions

In the preceding chapters we presented an approach to derive semi-autonomously reference information of a scenario from sensor data recordings including at least a camera and a distance measuring device. This reference data can be extended by further object properties, e. g. classification information and other object immanent attributes, and can be saved as an XML file along with the recordings. Additionally, we pointed out in which way this information can be used to assess the performance of a perception system by analyzing the proportion between the perception system's output and the reference data and to calculate performance measures. These performance figures can not only by consulted for evaluation purposes but may also be used to analyze and compare different parameter sets of algorithms as the runtime of the evaluation process is negligible (once the reference data has been extracted). Hence, this method also provides ideal support in the development phase of perception systems.

V. Future Work

As the presented methodology for evaluation is able to analyze the perception system's output regarding the existence of perception objects only and not to assess their position, orientation and dimension accuracy, it is planned to extend the presented method by an accuracy evaluation. In this regard the reference extraction process has to be at first analyzed in respect of the perception objects' precision.

The extracted reference information offers still more potential: Via the projection of (the adequate classification type of) reference objects into the camera image, sample image databases for the training of classifiers (cf. [7], [8]) can be generated automatically. Furthermore, it would be feasible to create an index of recordings from the reference information. This index would allow an easy and fast access to all scenes in an amount of recordings fulfilling some special constraints, e. g. pedestrian crossing the street in front of the ego vehicle from right to left.

VI. Acknowledgement

The work described within this paper is part of the results obtained in COMPOSE and ProFusion2, which are subprojects of the PReVENT Integrated Project, an automotive initiative co-funded by the European Commission under the 6th Framework Program. The long term goal of the subproject COMPOSE is to achieve measurable improvements in terms of severity of rear-end collisions and collision of vehicles with vulnerable road users by means of a collision mitigation application. In order to attain this goal, ProFusion2 develops innovative, robust and reliable sensor data fusion techniques.

The shown data screen-shots base on the BMW/FORWISS COMPOSE sensor fusion system from the BMW demonstrator (cf. [9]) and are by courtesy of COMPOSE. The presented reference data and statistics generation methods are integrated into the COMPOSE sensor fusion system and use its graphical interface [10].

References

[1] M. Bertozzi, A. Broggi, P. Grisleri, A. Tibaldi and M. Del Rose, "A tool for vision based pedestrian detection - performance evaluation," in *Proceedings of the IEEE Intelligent Vehicles Symposium 2004*, IEEE Press, Parma, Italy, June, 2006, pp. 784–789.

[2] D. Doermann and D. Mihalcik, "Tools and Techniques for Video Performance Evaluation," in *Proceedings of the IEEE Intl. Conference on Pattern Recognition*, Vol. 4, Barcelona, Spain, Sept., 2000, pp. 167–170.rt

[3] C. Jaynes, S. Weeb, R. M. Steele and Q. Xiong, "Development Environment for Evaluation of Video Surveillance Systems," in *Proceedings of the IEEE Intl. Workshop on Performance Analysis of Video Surveillance and Tracking*, Copenhagen, Denmark, June, 2002.

[4] K. Vogel, "Acquisition of High-Precision Reference Data for Evaluation of Active Safety Systems – Applicable of a RTK-GPS Surveying System," submitted to the *6th European Congress and Exhibition on Intelligent Transport Systems and Services*, Aalborg, Denmark, 2007.

[5] J. Black, T. Ellis and P. Rosin, "A Novel Method for Video Tracking Performance Evaluation," in *Proceedings of the Joint IEEE Intl. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, Nice, France, Oct., 2003.

[6] K. Fürstenberg and Jochen Scholz, "Reliable Pedestrian Protection using Laserscanners," in *Proceedings of the 8th Intl. Conference on Intelligent Transportation Systems*, Vienna, Sept., 2005.

[7] U. Scheunert, Ph. Linder, E. Richter, T. Tatschke, D. Schestauber, E. Fuchs and G. Wanielik, "Early and Multi Level Fusion for Reliable Automotive Safety Systems," submitted to the *IEEE Intelligent Vehicles Symposium 2007*, Istanbul, 2007.

[8] L. Walchshäusl, R. Lindl, "Multi-Sensor Classification using a boosted Cascade Detector," submitted to the *IEEE Intelligent Vehicles Symposium 2007*, Istanbul, 2007.

[9] L. Walchshäusl, R. Lindl, K. Vogel, T. Tatschke. "Detection of Road Users in Fused Sensor Data Streams for Collision Mitigation," in *Proceedings of Advanced Microsystems for Automotive Applications 2006*, J. Valldorf, W. Gessner, Springer, Berlin, April, 2006, pp. 53–65.

[10] R. Lindl and L. Walchshäusl, "Three-Level Early Fusion for Road User Detection," in *PReVENT Fusion Forum e-Journal*, 2006, pp. 19–24.