Identification of vehicle tracks and association to wireless endpoints by multiple sensor modalities

Daniel Becker*, Jens Einsiedler*‡, Bernd Schäufele†, Alexander Binder*§, Ilja Radusch†

* Fraunhofer Institut for Open Communication Technologies (FOKUS), Kaiserin-Augusta-Allee 31, 10589 Berlin, Germany
{daniel.becker, alexander.binder}@fokus.fraunhofer.de
† Daimler Center for Automotive Information Technology Innovations (DCAITI), Ernst-Reuter-Platz 7, 10587 Berlin, Germany
{bernd.schaeufele, ilja.radusch}@dcaiti.com
‡ Fraunhofer Application Center for Wireless Sensor Systems, Am Hofbräuhaus 1, 96450 Coburg, Germany
jens.einsiedler@iis.fraunhofer.de
§ Machine Learning Group, TU Berlin, Marchstr. 23, 10587 Berlin, Germany

***Abstract*—Vehicular positioning technologies enable a broad range of applications and services such as navigation systems, driver assistance systems and self-driving vehicles. However, Global Navigation Satellite Systems (GNSS) do not work in enclosed areas such as parking garages. For these scenarios, a wide range of indoor positioning technologies are available inside the vehicle (internal) and based on infrastructure (external).**

Based on our previous work, we use off-the-shelf network video cameras to detect the position of moving vehicles within the parking garage in multiple non-overlapping camera views. Towards the goal of using this system as positioning source for vehicles, detected positions need to be transmitted to the communication endpoint in the correct vehicle. The key problem thereby is the association of the externally-observed position to the endpoint in the corresponding vehicle. State-of-the-art tracking-by-detection techniques can differentiate multiple camera-detected vehicles but the generated tracks are anonymous and cannot inherently be associated to the corresponding vehicle.

To bridge this gap, we present a tracking-by-identification solution which analyzes vehicle movement patterns by multiple vehicle sensor modalities and compares them with camera-detected tracks to identify the track with the best correlation. The presented approach is based on Kalman Filters and suitable for real-time operation. Test results show that a correct and robust association between endpoints and camera-detected tracks is achieved and that occurring identity switches can be resolved.

I. Introduction

From navigation systems over driver assistance systems to self-driving vehicles, accurate positioning systems have become indispensable to modern vehicles. Common internal positioning systems use GNSS [1] to determine the absolute position of the vehicle. However, these systems have the disadvantage that they are inoperative in enclosed areas. In reality driving does not only stop in front of buildings but often continues into parking garages, tunnels, etc. Especially in urban areas an increasing number of multi-story car parks have been built. In order to provide a seamless integration between outdoor and indoor operation of the existing applications, additional indoor positioning systems are necessary.

From client-based SLAM [2], [3] over wireless positioning systems [4], [5] to vision based systems [6], [7], a wide range of technologies provide solutions for this task. Especially vision based systems represent a promising technology due to the favorable combination of accuracy, processing rate and system cost. These systems achieve sufficient accuracies in narrowly-spaced buildings and offer low system cost due to the ubiquity of cameras. However, one of the key problems of external vision based systems is the missing identity information from the detected vehicles in the camera images and thus difficult association of determined positions to the wireless endpoints in the vehicles.

Tracking targets detected across multiple non-overlapping camera views is a common challenge in computer vision which has been thoroughly investigated and many different solutions have been proposed [8]. Most of the approaches are based on the concept of *detection-by-tracking* [9], [10] which enables to differentiate multiple moving targets based on spatiotemporal information. However, if several targets come close to each other and disperse, identity switches are likely to occur which are not resolvable by these methods. Moreover, the tracked targets cannot be identified, i.e. they are anonymous.

Another class of algorithms is often called *tracking-by-detection* [11] as appearance-based information is incorporated. These methods are more complex but have the advantage that identity switches can be detected and resolved. In most cases the tracked targets are still anonymous since they can be differentiated according to their appearance but it is not possible to obtain a unique identification of the tracked target.

The third class of tracking methods is the so called *tracking-by-identification* [12] which combines the anonymous camera-based tracks with other positioning systems which include strong identity information, such as client-side positioning systems. Our approach falls into this category of *tracking-by-identification* techniques: We present a concept for the association of anonymous camera-detected tracks to communication endpoints in vehicles equipped with multiple sensor modalities. To this end, multiple track correlation metrics are defined which enable the comparison of camera-detected tracks and vehicle sensor modalities (e.g. odometry sensors, WiFi information, etc.). These metrics consider the vehicle's movement pattern (e.g. acceleration, speed, rotation etc.) which can be

compared with global movement patterns recorded by the cameras to find the track which constitutes the best match.

Based on our previous work [13], it is possible to reliably detect moving vehicles and to track them across multiple non-overlapping camera views in the context of parking garages. The central goal is to create a camera-based indoor localization system for vehicles which seamlessly takes over when GPS is unavailable. To achieve this goal, the externally-observed positions needs to be transferred to the communication endpoint in the corresponding vehicle. The proposed approach enables the identification of the camera track with the best correlation to the vehicle's sensor modalities. By creating an association between endpoint and the identified track, only appropriate camera-detected positions are forwarded to the endpoint.

The paper is organized as follows. In Section 2 related approaches are presented. The system overview of all components is provided in Section 3. In Section 4, the methodology is provided, that represents the essential building blocks for the system. Section 5 contains the experimental evaluation in a realistic scenario. The paper closes with a conclusion and future outlook in Section 6.

II. Related Literature

This work can be considered as extension of the *eValet* [13] project which is an indoor micro-navigation system for parking garages based on off-the-shelf network cameras. In this project, infrastructure-based cameras are utilized to determine the global position of moving vehicles. Additionally, indoor maps of the parking garage are provided as well as information about free parking lots.

Another project [14] employs infrastructure-based LIDAR scanners inside carparks in order to enable autonomous self-driving vehicles. The laser scanners fulfills the same task as the cameras in the *eValet* project which is the external detection and positioning of objects within the parking garage. However, laser scanners are more expensive and more limited in the quantity of provided information compared to cameras.

A camera-based detection and tracking system for tunnels is proposed in [15]. The goal is to reliably keep track of vehicles even under harsh operational conditions such as low illumination and a high density of vehicles. A subset of detected vehicle features is provided to the tracking algorithm which represents a tracking-by-detection approach.

Although infrastructure cameras are widely-deployed in public parking areas (e.g. surveillance cameras), there are relatively few attempts in using these cameras as positioning system for vehicles. In the case of pedestrians however, an external camera-based positioning system is presented in [12]. In this system, infrastructure cameras are used to detect multiple persons which are equipped with a Ultra-Wide-Band (UWB) radio positioning system. A tracking-by-identification approach is used to reliably track the individual persons and to maintain their identity and resolve identity switches.

To the best of our knowledge, there is no comparable approach utilizing multiple vehicular sensor modalities for identifying tracked targets across multiple non-overlapping camera views. Moreover, the utilization of commercial cameras to perform seamless indoor-positioning for vehicles in parking garages represents a unprecedented application scenario.

III. System Overview

The infrastructure side of our camera-based positioning system is partitioned into three software components as shown in Fig. 1: The *Detection Module*, the *Tracking Module* and the *Identification Module*. Vehicles are also equipped with a software component, our so called *InCarPlatform* which can be considered as positioning client and sensor information provider to the positioning system.

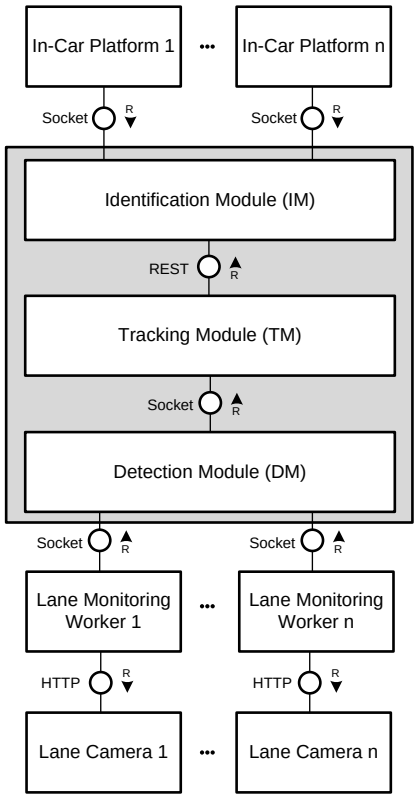

Fig. 1. FMC diagram of the system.

A detailed explanation of each component and their interaction is provided subsequently:

A. Detection Module

We utilize customary monocular network cameras to monitor the lanes within the car park. For this purpose, each camera's video stream is analyzed by a dedicated software agent. Our so called *Lane Monitoring Worker* (*LMW*). Once a *LMW* detects a vehicle, the software calculates the geo-position (latitude, longitude and elevation) of the vehicle and sends it to the *Detection Module* (*DM*) which takes over the management of the *LMW* and provides the geo-position information to the *Tracking Module* (*TM*).

The detection is implemented in a three stage approach. During the first stage, a combined motion detection and classification algorithm locates all moving vehicles within the camera field of view. During the next step, features of the detected vehicles are extracted and matched to features of already known ones to recognize new vehicles in the camera view. In this way, the vehicles are tracked from their entrance in the camera-observed area until they leave. Based on the relative distance of the vehicle to the camera and the a priori knowledge of the mounting location of the camera the geo-position of each vehicle is calculated and sent to the *DM* in the last stage.

During this three step process, each *LMW* works independently and delivers the information to the *DM* which forwards it directly to the *Tracking Module*.

B. Tracking Module

The task of this component is the aggregation of individual camera-detected positions to tracks. Incoming camera-detected positions from the *DM* are combined into tracks according to spatiotemporal information, thus representing a detection-by-tracking approach. Thus, a track consists of a series of camera-detected points and a TID (track identifier). Furthermore, tracks are managed dynamically, i.e. new tracks can be created and outdated tracks can be removed after a certain timeout. Ideally, a single track should represent a single camera-detected target. Also, the TM has the task of filtering clutter, e.g. due to misdetections. In contrast to the local tracking in image space performed by the *DM*, the tracking performed by the *TM* is referred to as global tracking as it takes place in object space and includes all camera views.

C. Identification Module

This component is in the main scope of this paper performing the identification of anonymous camera-detected vehicle tracks based on multiple sensor modalities in the vehicle, thus representing a tracking-by-identification approach. The identified tracks are associated to the communication endpoint in the corresponding vehicle, which enables the transmission of the correct externally-observed positions towards the goal of serving as positioning source for the vehicle.

In order to use the infrastructure camera positioning service, vehicles register at the *IM* which then sends camera-detected positions to the appropriate registered endpoint. In order to check if a camera-detected position belongs to a registered vehicle, multiple sensor modalities of the vehicle are utilized to reconstruct its movement pattern. In other words, the *IM* forwards each incoming camera-detected position to corresponding vehicle endpoints based on the correlation of the vehicle's sensor modalities.

D. In-car Platform

This software component is deployed inside of the vehicle and represents a positioning client which registers at the *IM*. Sensor modalities from the vehicle are provided to the *IM*, which in turn performs an identification of the corresponding camera-detected track and transmits the correct positions. Usually a computer integrated in the vehicle or a smartphone is suitable to run the in-car platform software component. The communication to the infrastructure components is established using wireless communication interfaces, e.g. 802.11 WiFi [16]. Additionally, a user interface is provided for displaying indoor maps and providing navigation services.

IV. Methodology

In the following, the main building blocks are provided in order to achieve the proposed vehicular positioning system based on external infrastructure cameras.

A. Camera Detection and Position Estimation

The detection of moving vehicles within the camera views is the objective of the *DM*. As mentioned forgoing, we use a three stage approach to detect and localize moving vehicles within the car park:

1) Vehicle Detection: First of all, the video stream is analyzed for moving objects to limit the subsequent object detection to this regions. For this purpose, we make use of an existing motion template algorithm implementation based on the work of G. Branski and J. Davis [17] which uses a Motion History Image (MHI) [18] to detect and track movements.

The MHI are updated in each processing cycle one time. During this update, all motions older than approximately 0.5s (= *fps*/2 frames) were deleted and the motions in the MHI were segmented into regions of interests (ROI). Additionally, we pre-filter the resulting list of ROIs in advance of the following object detection by rejecting all elements whose dimensions are smaller than 10% of the whole image size to reduce the computational time. Adjacent or overlapping segments are also being bound together.

As Fig. 2 illustrates, the motion detection provides region of interests (Fig. 2, dashed rectangle) and the objects direction of movement (Fig. 2, dashed arrow).

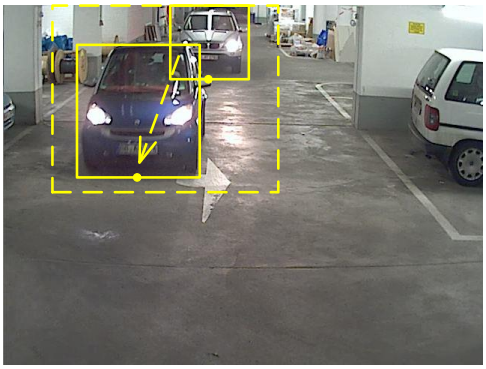

Fig. 2. Detected driving vehicles with root points.

Detecting objects in static images is a common challenge and accordingly a well investigated area of the computer

vision. For example, C.P. Papageorgiou and T. Poggio demonstrate successfully in [19] that the known approaches from other domains can be transferred to detect vehicles.

For our solution, we use a cascade of boosted haar-like feature classifier. This approach was proposed for the first time by P. Viola and M. Jones [20] and has been improved by R. Lienhart and J. Maydt [21]. We trained our classifier with approximately 3500 positive and negative training images, taken in a prototypical equipped parking garage.

As figure 2 illustrates, the motion and object detection in this stage provides a bounding box of the detected vehicles (see figure 2, rectangle).

2) Local Tracking: After the moving object in the region of interest was classified as a vehicle, the local identification process starts. For this purpose, Speeded Up Robust Features (SURF) [22] of each vehicle are extracted and mathematically transformed into feature descriptors to make them comparable. Each identified vehicle holds a set of these descriptors.

In each frame where a moving object is detected the computed descriptors are compared to the set of already known ones using the so called Brute Force Matcher. It tries to find the nearest neighbors of all descriptors from the current frame to the known descriptor set [23]. If there are enough of these nearest neighbor matches the object is reidentified. Otherwise, it was an unknown/new vehicle and will be added to the list of known objects.

3) Position Determination: The relative distance between vehicles and camera will be determined based on interpolation of distances from a coarse rectangular grid laid out in the viewable of each camera with known distances between grid points and to the camera.

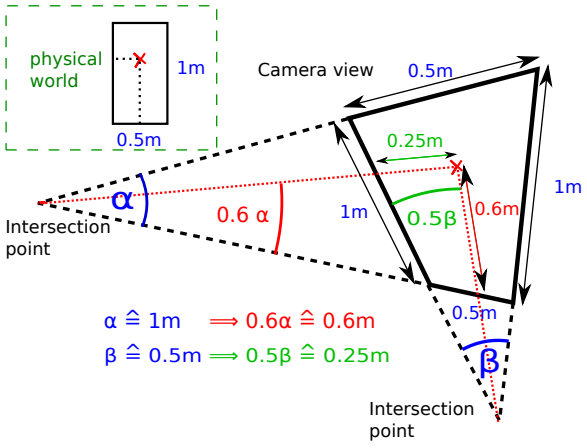

Fig. 3. Geometrical principle for interpolation of grid coordinates in the non-Cartesian camera view. Grid is overly distorted for better visualization.

A grid is oriented parallel to the midline of a one-way lane in the parking garage and orthogonal to it. Points are placed with a distance of $0.5m$ orthogonal to the midline of a lane, and with $1m$ distance parallel to the midline, usually covering distances of $4m$ to $20m$ from the camera in the direction parallel to the midline of a lane. The coordinates of the rectangular grid are non-Cartesian in the camera view. The interpolation algorithm is based on the assumption that geodesics in the camera view are still lines, though. The interpolation is done by computing the intersection points between two lines of two opposing edges for each cube in the grid and translating angles into relative distances as shown in Fig. 3. This yields precise distances also for points outside the grid, particularly in the direction orthogonal to the midline.

B. Global Tracking

The global tracking of all vehicles in object space spanning all camera views is the objective of the *TM*. For each vehicle, the tracking is based on a 5-tuple of state variables $X := (x, y, v, \psi, \omega)$ consisting of spatial positions of the vehicle (x, y) in UTM coordinates, its heading angle ψ, its speed in tangential direction v and the differential change ω of the heading angle Ψ. We assume a non-linear evolution of the state variables from one time step to the next given by equation (1).

$$a(x, y, v, \psi, \omega) = \begin{pmatrix} x + v\,\Delta t \sin(\psi) \\ y + v\,\Delta t \cos(\psi) \\ v \\ \psi + \Delta t\,\omega \\ \omega \end{pmatrix} \tag{1}$$

Each of the state variables corresponds to an observed variable taken from odometry or camera measurements.

The dynamics of these state variables are modeled within the Bayesian filtering framework which under assumption of gaussian noises yields the equations (2) and (3).

$$a(X_{k+1}) = a(X_k) + \epsilon_k,\ \epsilon_k \sim N(0, Q_k) \tag{2}$$

$$Y_{k+1} = H_{k+1} X_{k+1} + \eta_{k+1}\ \eta_{k+1} \sim N(0, R_{k+1}) \tag{3}$$

The covariances Q_k, R_k of the gaussians are diagonal. The relation between observed variables Y and the state variables in X in equation (3) is linear with gaussian noise η and a diagonal matrix H with 0/1-entries. Zero entries in the diagonal of H mask measurements which are not observed at the current time step. We use extended Kalman filtering [24] to resolve the nonlinearity with respect to position coordinates x, y in equation (1). Since the relationship in equation (3) between observed and state variables is still a linear one, the difference to a conventional Kalman filter is the usage of the linearized update A_k as given in equation (5) for the predicted covariance estimate matrix P_{k+1}^- from equation (7). Alternatives are unscented Kalman filtering [25] and Monte Carlo methods [26].

$$\frac{\partial a}{\partial X} = \begin{bmatrix} 1 & 0 & \Delta t \sin(\psi) & v\,\Delta t \cos(\psi) & 0 \\ 0 & 1 & \Delta t \cos(\psi) & -v\,\Delta t \sin(\psi) & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$A_k = \left.\frac{\partial a}{\partial X}\right|_{X=\hat{X}_k} \tag{5}$$

For the sake of completeness, the remaining KF equations are provided: A priori state estimate (6) and covariance estimate (7), Kalman Innovation (8), Kalman Gain (9), a posteriori state estimate (10) and covariance estimate (11).

$$X_k^- = a(X_{k-1}) \tag{6}$$

$$P_k^- = A_k P_{k-1} A_k^T + Q_k \tag{7}$$

$$N_k = Y_k - H_k X_k^- \tag{8}$$

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \tag{9}$$

$$X_k = X_k^- + K_k N_k \tag{10}$$

$$P_k = (I - K_k H_k) P_k^- \tag{11}$$

For each detected track one generalized Kalman filter will be created. A heuristic is used to create Kalman filters from outlier detections by the cameras if a number of detections are found to be close to each other within a certain time frame and distance but far from all existing Kalman filter predictions. Similarly Kalman filters will be deleted once they are not supported by data for a certain time window.

C. Tracking-by-identification Approach

This component represents the main scope of this paper which resides in the *IM*. Multiple sensor modalities are utilized from each registered vehicle in order to compare its movement pattern to the camera-detected tracks and find the one with best correlation. Thus, several *track correlation metrics* are defined which indicate the correlation of camera-detected tracks and sensor modalities. Fig. 4 illustrates the methodology: For each registered vehicle, a *Track Identification* instance is created which contains one tracking filter for each camera track.

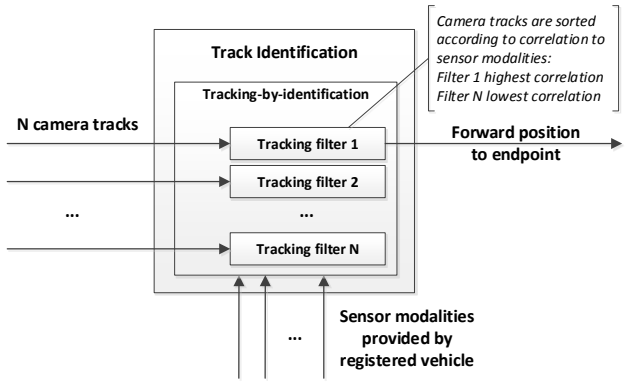

Fig. 4. Overview about tracking-by-identification methodology.

The tracking filters are sorted according to their correlation to the sensor modalities of the vehicle, i.e. the first track has the highest and the last track the lowest correlation respectively. Hence, the first track is considered belong to the registered vehicle. Consequently, only incoming camera-detected positions to the first track are forwarded to the actual registered endpoint whereas incoming positions to other tracks are disregarded.

1) Tracking filter: The tracking filter is an EKF as described in Section IV-B. A separate tracking filter is used for each camera-detected track.

2) Suitable Sensor Modalities: The fundamental idea is to use vehicle sensor modalities which measure the movement and position of the vehicle in order to derive metrics to achieve a comparison with the camera-detected tracks. Basically, a classification into absolute and relative sensor modalities is sensible. Absolute sensor modalities can be derived from sensors measuring position and driving direction of the vehicle (e.g. WiFi positioning and magnetic compass resp.). Relative sensor modalities on the other hand, can be acquired from sensors measuring the velocity or angular velocity of the vehicle (e.g. wheel speed sensor and wheel angle sensor resp.).

3) Track Correlation Metrics: In the following, several metrics are defined to assess the correlation of camera-detected tracks to absolute and relative sensor modalities respectively. Absolute sensor modalities provide position or heading information for the vehicle which can be compared to the current position and heading of all camera-detected tracks. In terms of position, the Euclidean distance between the vehicle's self-position ($x_{s,k}$,$y_{s,k}$) (e.g. obtained by WiFi positioning) and the current externally-observed position of each camera track (x_k,y_k) at each time step k yields metric M_3:

$$M_3 = \sqrt{(x_k - x_{s,k})^2 + (y_k - y_{s,k})^2} \tag{12}$$

In order to improve the accuracy and robustness of the vehicle's self-position estimation, two additional state variables are added to the EKF state transition matrix in (1): $x_s = x_s + v\Delta t \sin(\psi)$ and $y_s = y_s + v\Delta t \cos(\psi)$. Hence, incoming measurements of the self-position (e.g. WiFi positioning) are assigned to x_s and y_s. In the following, the a posteriori state estimate of x_s and y_s at time step k is referred to as Kalman-filtered self-position ($x_{sf,k}$,$y_{sf,k}$).

Consequently, metric M_4 is defined as the Euclidean distance between the Kalman-filtered self-position ($x_{sf,k}$,$y_{sf,k}$) and the camera-detected position (x_k,y_k):

$$M_4 = \sqrt{(x_k - x_{sf,k})^2 + (y_k - y_{sf,k})^2} \tag{13}$$

In terms of the heading ψ, the absolute value of the angle difference between a measured heading inside the vehicle $\psi_{s,k}$ and camera-observed heading ψ_k is considered as metric M_5:

$$M_5 = |\psi_k - \psi_{s,k}| \tag{14}$$

In this case, the lowest correlation occurs for two vehicles which are driving in opposite directions i.e. having an angle difference of 180 degrees.

For relative sensor modalities, track correlation metrics are derived by utilizing internal properties of the tracking filter, i.e. the *Kalman Innovation* $N_{1,k} = N_k$ from (8). $N_{1,k}$ represents the difference between measurements and predictions for each state variable and thus reflects the correlation between the relative sensor modalities (e.g. speed, wheel angle) and camera-

detections. However, $N_{1,k}$ is not suitable for deriving track correlation metrics because it is prone to variations of noise in different camera detection areas. For instance, areas with good lighting conditions and thus highly accurate camera detections might always have the best correlation regardless of the real track identity. To overcome this limitation, the smoothed state estimate of the KF $X_k = X_{1,k}$ is fed into a second KF which has near-zero measurement covariance matrix R_k, thus the state estimate $X_{2,k}$ instantly converges to $X_{1,k}$. Hence, the *Kalman Innovation* at the second KF results in:

$$N_{2,k} = H_k X_{1,k} - H_k X_{2,k}^- \tag{15}$$

$N_{2,k}$ can be simplified: Assuming that $X_k = X_{2,k} = X_{1,k}$ and considering (10), $X_k = X_k^- + K_k N_{1,k}$ is substituted into (15) obtaining:

$$N_{2,k} = H_k K_k N_{1,k} \tag{16}$$

Hence, track correlation metrics are derived from $N_{2,k}$. By performing an averaging over the last D values, the movement of the vehicle over a certain time period is considered. The number of values D depends on the KF update interval ΔT_U and on the averaging time interval ΔT_{avg}:

$$D = \frac{\Delta T_{avg}}{\Delta T_U} \tag{17}$$

Hence, the metrics M_1 and M_2 are defined as follows:

$$M_1 = \frac{1}{D} \sum_{i=k-D}^{k} \sqrt{N_{2,x,i}^2 + N_{2,y,i}^2} \tag{18}$$

$$M_2 = \frac{1}{D} \sum_{i=k-D}^{k} |N_{2,\psi,i}| \tag{19}$$

Assuming $\Delta T_U = 100ms$ and $\Delta T_{avg} = 5000ms$, then $D = 50$ i.e. M_1 and M_2 comprise 50 values.

A simplified illustration of the track correlation metrics is shown in Fig. 5. For relative sensor modalities, measurements and predictions are compared, thus $N_{2,k}$ is expressed by its components, c.f. (16): $N_{2,x,k} \mathrel{\widehat{=}} x_k - x_k^-$, $N_{2,y,k} \mathrel{\widehat{=}} y_k - y_k^-$ and $N_{2,\psi,k} \mathrel{\widehat{=}} \psi_k - \psi_k^-$. In comparison to the metrics M_3, M_4 and M_5 derived from absolute sensor modalities, M_1 and M_2 are also based on the Euclidean distance and angle difference resp. The essential difference is that the average aberration of incoming camera-detected positions to the KF a priori estimate is evaluated, instead of absolute aberrations.

D. In-car Platform

The intention of the *In-car Platform* is to provide an indoor car park navigation system that resembles GPS navigation systems for the public road network [13]. For this purpose precise positioning is required, which is available through the external camera-based positioning system.

For the connection between the infrastructure and the vehicles, customer WiFi based on the 802.11g standard [16] is used. For communication over 802.11g WiFi a proprietary protocol is used. The *In-car Platform* opens a TCP socket, registers at the *IM* by accessing a registration webpage where

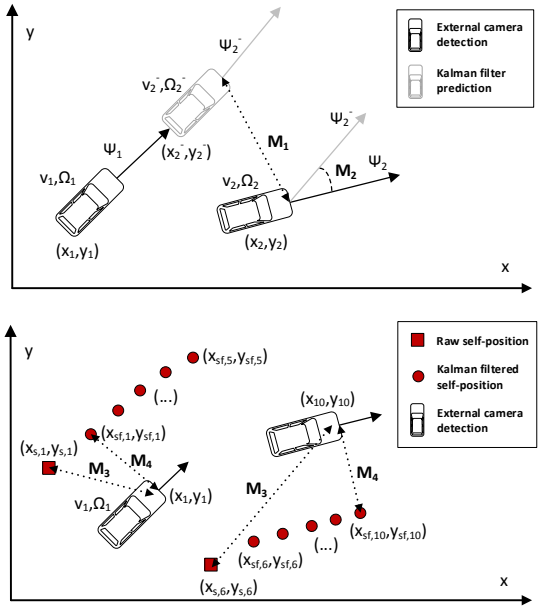

Fig. 5. Identification metrics M_1 and M_2 based on relative sensor modalities (top) and M_3 and M_4 based on absolute sensor modalities (bottom).

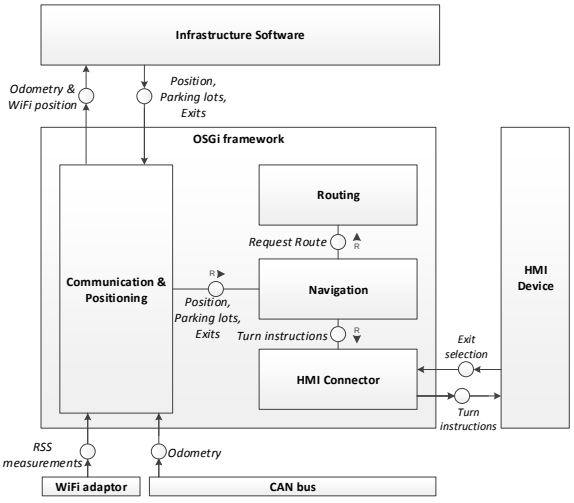

Fig. 6. FMC diagram of the in-car platform.

the number of the open TCP port is submitted. Subsequently, the *IM* establishes a TCP connection to the *In-car Platform* which is used to provide vehicular sensor modality data (e.g. WiFi positioning, odometry). In turn, the *In-car Platform* receives position data from the *IM* once the vehicle is recognized in the camera views and the correct track is assigned.

The in-car platform consists of several components referred to as bundles, as can be seen in Fig. 6. The Java-based OSGi [27] framework has been selected as platform because it offers advanced modularization capabilities.

The *Communication & Positioning* bundle is intended for connections to external components and for the management

of positions. On the one hand, sensor modalities from the vehicle are accessed. Received Signal Strength (RSS) measurements for a WiFi positioning technique and odometric sensor data are acquired from the WiFi adaptor and CAN bus respectively. Given a set of RSS measurements, a WiFi position is estimated based on the approach presented in [28]. On the other hand, communication with the *IM* on the infrastructure side is performed, in order to provide sensor modalities and receive positioning data.

The *Communication & Positioning* bundle forwards the positioning data to the *Navigation* bundle. The latter provides a turn-by-turn navigation that directs the driver to a free parking lot. Thereby it makes use of the *Routing* bundle which has a database with the way network of parking garages that the driver uses regularly. Moreover, for unknown parking garages it retrieves map data from an appropriate server.

Last but not least, a HMI represents the connection to the driver for displaying map and navigation instructions.

V. Evaluation

For our project, we have a common company parking garage equipped with both network cameras to observe the lanes and a WiFi infrastructure. The test site within the parking garage has a total area of around $2800m^2$ and includes an overall lane length of appr. $350m$. The road layout has two long straight roads with three joints, as shown in figures of Table V-B.

The lanes are monitored by six network cameras (AXIS Q1604). The cameras provide an MJPEG encoded video stream with $1280x720px$ and $24fps$. The illumination within the camera monitored area differs between $40lx$ and $240lx$. The WiFi infrastructure consists of nine APs (TP-Link TL-WA901ND with OpenWRT OS) evenly distributed to provide car-to-infrastructure communication and WiFi positioning.

Evaluation results for the utilized sensor modalities, the temporal behavior of all data flows and the tracking-by-identification approach are provided subsequently.

A. Sensors

A dynamic WiFi-positioning system [28] has been implemented in the parking garage, which has an accuracy of 10m in 90% of all estimations. In the current implementation, about every $2.5s$ a new position is estimated. The position is estimated at the vehicle and transmitted to the *IM* via WiFi.

As described in detail in [13], we are able to provide a higher positioning accuracy in comparison to standard GPS [29] with our camera-based indoor positioning system. As described in Section V-B, we can ensure a positioning update rate of more than $10Hz$ due to several improvements compared to the system in [13].

Odometric data (e.g. vehicle speed and wheel angle) can be accessed at the CAN bus at modern vehicles which use these information for a wide range of driver assistance applications. Recent Field Operational Tests (FOT) for V2X applications [30] demonstrate the utilization of odometric sensor data. In our experiments, a *Smart Fortwo* has been used (see Fig. 2) providing odometric data over a CAN bus adaptor which is transmitted to the *IM* via WiFi.

B. Temporal Behavior

According to [31], we developed a testing tool to measure the *capture-to-LMW* delay, i.e. the time that elapses from capture to provision of the images to the *LMWs*. The *LMW* is implemented in C++ and uses the computer vision library OpenCV (v2.4.5). The following delay measurements were performed on a notebook with an Intel(R) Core(TM) i7 (2860QM 2.5GHz) and 16GB of RAM on Ubuntu 12.04 LTS (64 bit) operating system. For MJPEG coded video streams, we measured a median delay of approximately $128ms$. The minimum delay was $89.39ms$ and the maximum delay $185.89ms$. In addition to the *capture-to-LMW* delay, the *LMW* requires between $6ms$ and $48ms$ to process an incoming frame until it can provide some information to the *DM*. In the first case, there are no changes within the image stream. In the second case, the *LMW* have to classify multiple movable objects within the camera observed area. The overall delay from image capturing to the provision of positioning information to the *TM* is between appr. $100ms$ and appr. $240ms$.

The *TM&IM* is implemented as Java Tomcat web application and its performance measured using the Java profiler *VisualVM* (v1.3.5). For the experimental scenarios in the following section, CPU and memory utilization are analyzed. The experiments are conducted on a desktop PC with an Intel(R) Core(TM)2 Duo CPU (E6750 2.66GHz) and 3 GB of RAM on Windows 7 (32bit) OS with Java JDK v1.7.0-21. The profiler results show a maximum CPU and heap memory utilization of 55% and 110MB respectively. Peaks in CPU utilization coincide with garbage collector (GC) activity thus an adjustment of the GC in productive applications is advisable. The *TM&IM* processing delay, i.e. the time between receiving a position detection from the *DM* until forwarding it to the *In-Car Platform*, is less than $10ms$ in 98% of all test cases.

On the *In-Car Platform*, the Network Time Protocol (NTP) [32] is used to achieve synchronous clocks with the infrastructure. The highest delay variation on the vehicle side is caused by the WiFi channel. For 97% of all transmission between infrastructure and vehicle (and vice versa) the delay is below $40ms$. The average delay is approx. $23ms$ and the maximum delay reaches $850ms$, mainly due to handoffs between APs.

Last but not least, the *Detection to Endpoint* delay is determined, i.e. the complete time period from image acquisition in the camera until reception of the detected position in the vehicle endpoint. The minimum, maximum and average *Detection To Endpoint* delay is $42ms$, $998ms$ and $150ms$ resp. The high maximum delay occurrs only in very few cases and can be attributed mainly to the WiFi transmission delay. By applying dead reckoning at the *In-Car Platform*, received positions can be advanced according to the current vehicle speed and direction, in order to mitigate the positioning error caused by the *Detection to Endpoint* delay.

Track	Vehicle speed	Duration	Trail
Z	3.0 .. 4.5 m/s	24 s	
E	2.3 .. 3.0 m/s	14 s	
F	2.6 .. 3.6 m/s	16 s	
G	0.0 .. 3.0 m/s	21 s	

TABLE I
OVERVIEW TEST SCENARIOS.

C. Tracking-by-identification Approach

We have developed a testing framework which enables the replay of recorded log files into the *IM* implementation in real-time. This framework is referred to as *Simulated Vehicle*, as it behaves like an actual vehicle, i.e. performing registration, providing sensor data and camera-detected positions. Moreover, multiple camera tracks can be replayed simultaneously. The advantage of this approach is that no modifications are required to the production system and that the dynamic behavior for complex scenarios can be evaluated realistically.

Table V-B shows four different camera-detected tracks with a track ID, vehicle speed, total duration and a small image showing the travelled path of the respective vehicle. These tracks represent vehicles driving on different paths with varying speeds and are used as input data for the testing framework, in order to simulate realistic scenarios. In the following experiments, track Z represents the correct camera track corresponding to the registered vehicle, whereas tracks E, F and G belong to other vehicles. Moreover, recorded data from WiFi positioning (absolute) and odometric (relative) vehicular sensor modalities correlating to track Z is provided to the *IM*. Consequently, the ground truth is known as vehicular sensor modalities and track Z are correlated.

As decision criterion for track identification, the metrics M_1, M_2 and M_4 are taken into consideration: A specific track is identified if at least two out of the three metrics indicate the best correlation for this track compared to all currently available tracks. A metric shows the best correlation for a specific track, if it has the lowest value of distance (M_1, M_4) or angle (M_2). To assess the track identification performance, the following metrics are defined: Given a total time period when the correct track is identified ΔT_c and one with incorrect identification ΔT_f, the *correct track identification rate* is defined as $\rho = \frac{\Delta T_c}{\Delta T_c + \Delta T_f}$. Moreover, the *maximum duration of incorrect track assignment* $\Delta T_{f,max}$ is defined as the maximum time interval of incorrect track identification.

In the following, several experiments are presented where the track correlation metrics are evaluated by applying the previously introduced testing concept.

Fig. 7 shows the results of Experiment 3 illustrating the setup for a scenario with four vehicles: The registered vehicle corresponds to track Z which is denoted as C and the three other vehicles (tracks E, F, G) as F1, F2 and F3 respectively.

The resulting cumulative distributions for metrics M_1, M_2 and M_4 (Fig. 7, top) show the strongest correlation for the correct track C which converges faster compared to the uncorrelated tracks F1, F2 and F3. Also, the timeline (Fig. 7, bottom) illustrates that each metric identifies track C for the majority of time steps. Interestingly, the combined metric (M_1 M_2 M_4) shows overall better results than each individual metric as the metrics appear to have complementary properties. Only once, for a time period of less than one second, the track F1 is incorrectly identified. Moreover, during the time period from 11.5s to 12.5s, the combined metric does not yield a clear result in terms of track identification as each metric identifies a different track (F3,C,F1). If this uncertainty period is considered as error, then ρ results in 93.11% and $\Delta T_{f,max}$ is 0.82s. However, it is possible to not consider this period as error as the correct track C was correctly identified before. Thus, in similar uncertain situations, the best strategy is to keep the previous association until the metrics become consistent. In this case, ρ is 97.39% and $\Delta T_{f,max}$ 0.61s.

Another scenario is presented in Experiment 1: An overtaking situation with two vehicles simulated by using camera track G (denoted as D) in combination with track Z (denoted as C). Track C has a higher vehicle speed than track D and both tracks are on the same path but track D starts with an offset, hence the overtaking maneuver occurs. Even though some metrics show a partially incorrect track identification, the combined metric almost always provides the correct result. Thus, ρ is 99.13% and $\Delta T_{f,max}$ is 0.089s.

Experiment 2 is identical to Experiment 1 with one exception: At the time of 7s after starting the experiment, the track identifiers of both tracks are swapped which constitutes an identity switch. Thus, it is important to consider that the camera track C is only the correct track for the first 7s of the experiment, until the identity switch occurs. After that, for the remaining 16s, track D corresponds to the registered vehicle and its sensor modalities. Fig. 8 shows the timeline chart where the identity switch is explicitly marked, i.e. it is also incorporated in the calculation of the correct track identification rate ρ. It takes approx. 2s to detect and to resolve the identity switch based on the combined metrics. Consequently, ρ is 90.91% and $\Delta T_{f,max}$ is 2.03s.

A summary of all conducted experiments is provided in Table V-C. All scenarios exhibit a correct track identification rate ρ between 90% and 100%. Even for the particularly challenging Experiment 2 with the occurring identity switch, a ρ of 90.91% is achieved and it takes 2.03s to detect and recover from the identity switch. Last but not least, a complex scenario with four vehicles is presented in Experiment 3 which demonstrates the viability of the methodology and the applicability of the track identification metrics on multiple camera tracks.

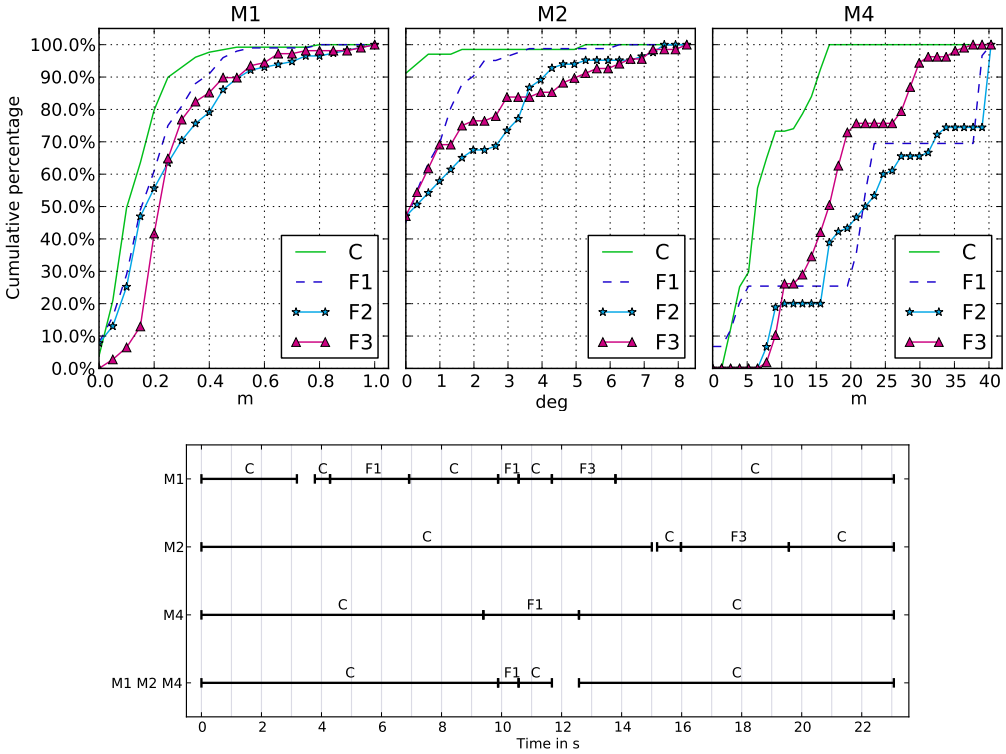

Fig. 7. Experiment 3 with four vehicles, cumulative track correlation metrics (top) and identified tracks per metric timeline (bottom).

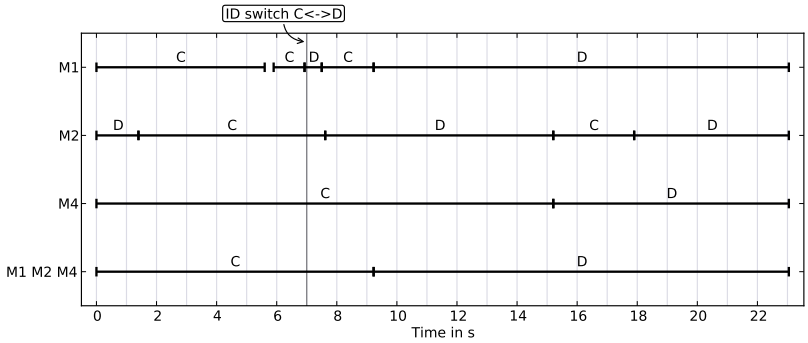

Fig. 8. Experiment 2 with two vehicles in overtaking scenario and ID switch, identified tracks per metric timeline.

Exp. number	Description	ρ	$\Delta T_{f,max}$
1	Overtaking scenario of two vehicles	99.13%	0.09s
2	Overtaking scenario of two vehicles with identity switch	90.91%	2.03s
3 A	Complex scenario with four vehicles on different partially crossing paths	93.10%	0.82s
3 B	Same as 3 A, but without considering "uncertainty" period as error	97.39%	0.61s

TABLE II
SUMMARY OF EXPERIMENTAL RESULTS.

VI. SUMMARY AND OUTLOOK

The presented tracking-by-identification methodology performs the identification of anonymous camera-detected vehicle tracks based on multiple sensor modalities provided by the vehicle. The identified tracks are associated to the registered communication endpoint in the corresponding vehicle which in turn continuously provides measurements of multiple sensor modalities. Additionally, the registered endpoints are positioning clients that receive their corresponding, externally-observed position via a wireless communication channel.

In order to identify the anonymous camera tracks, track correlation metrics based on absolute and relative sensor

modalities are introduced: Relative sensor modalities describe the movement pattern of the vehicle (e.g. odometric data) and absolute sensor modalities provide information about the absolute states of the vehicle (e.g. WiFi positioning). Based on the defined track correlation metrics, a detailed experimental evaluation has been conducted resulting in correct track identification rates of at least 90% for the investigated scenarios. Additionally, in one of the test scenarios, an identity switch of two camera tracks is simulated which is successfully detected and resolved within approx. 2s. Hence, the experimental results confirm the viability and robustness of the proposed tracking-by-identification approach.

In terms of future work, further experiments with additional sensor modalities in the vehicle can be conducted, towards the goal of complementing the existing metrics and achieving an even more robust overall track identification. To this end, different methods of combining the individual track correlation metrics need to be explored, in order to derive a combined metric which yields an optimal correct track correlation rate for a set of common test scenarios.

Another research direction is the integration of the proposed methodology with other existing positioning systems. Especially the integration with state-of-the-art smartphones could be beneficial in several aspects: On the one hand, smartphone sensors can be utilized as sensor modalities for the tracking-by-identification module. On the other hand, the smartphone can act as positioning client receiving the highly accurate camera-detected positions.

REFERENCES

[1] Elliott D Kaplan and Christopher J Hegarty. *Understanding GPS: principles and applications*. Artech House Publishers, 2006.

[2] V. Nguyen, A. Harati, A. Martinelli, R. Siegwart, and N. Tomatis. Orthogonal slam: a step toward lightweight indoor autonomous navigation. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 5007–5012, 2006.

[3] R. Kummerle, D. Hahnel, Dmitri Dolgov, S. Thrun, and W. Burgard. Autonomous driving in a multi-level parking structure. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 3395–3400, 2009.

[4] Hui Liu, H. Darabi, P. Banerjee, and Jing Liu. Survey of wireless indoor positioning techniques and systems. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 37(6):1067–1080, 2007.

[5] Yanying Gu, A. Lo, and I. Niemegeers. A survey of indoor positioning systems for wireless personal networks. *Communications Surveys Tutorials, IEEE*, 11(1):13–32, 2009.

[6] S. Heissmeyer, L. Overmeyer, and A. Muller. Indoor positioning of vehicles using an active optical infrastructure. In *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*, pages 1–8, 2012.

[7] Chih-Jen Wu and Wen-Hsiang Tsai. Location estimation for indoor autonomous vehicle navigation by omni-directional vision using circular landmarks on ceilings. *Robotics and Autonomous Systems*, 57(5):546–555, 2009.

[8] E. Trucco and K. Plakas. Video tracking: A concise survey. *Oceanic Engineering, IEEE Journal of*, 31(2):520–529, 2006.

[9] K. Otsuka and N. Mukawa. Multiview occlusion analysis for tracking densely populated objects based on 2-d visual angles. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–90–I–97 Vol.1, 2004.

[10] P. Nillius, J. Sullivan, and S. Carlsson. Multi-target tracking - linking identities using bayesian network inference. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2187–2194, 2006.

[11] H. Ben Shitrit, J. Berclaz, F. Fleuret, and P. Fua. Tracking multiple people under global appearance constraints. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 137–144, 2011.

[12] Rok Mandeljc, Stanislav Kovačič, Matej Kristan, Janez Perš, et al. Tracking by identification using computer vision and radio. *Sensors*, 13(1):241–273, 2012.

[13] Jens Einsiedler, Oliver Sawade, Bernd Schäufele, Marcus Witzke, and Ilja Radusch. Indoor micro navigation utilizing local infrastructure-based positioning. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 993–998. IEEE, 2012.

[14] André Ibisch, Stefan Stümper, Harald Altinger, Marcel Neuhausen, Marc Tschentscher, Marc Schlipsing, Jan Salmen, and Alois Knoll. Towards autonomous driving in a parking garage: Vehicle localization and tracking using environment-embedded lidar sensors. 2013.

[15] Reyes Rios Cabrera, Tinne Tuytelaars, and Luc Van Gool. Efficient multi-camera detection, tracking, and identification using a shared set of haar-features. In *Proceedings IEEE computer society conference on computer vision and pattern recognition-CVPR2011*, 2011.

[16] *IEEE 802.11g. Further higher-speed physical layer extension*, 2003.

[17] G. Bradski and J. Davis. Motion segmentation and pose recognition with motion history gradients. In *IEEE Workshop on Applications of Computer Vision 2000 (WACV 2000)*, pages 238–244, 2000.

[18] A. Bobick and J. Davis. Real-time recognition of activity using temporal templates. In *IEEE Workshop on Applications of Computer Vision 1996 (WACV 1996)*, pages 39–42, 1996.

[19] C.P. Papageorgiou and T. Poggio. A trainable object detection system: Car detection in static images. *AI Memo, no. 1673, CBCL Paper No 180, Massachusetts Institute of Technology*, 1999.

[20] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE International Conference on Computer Vision and Pattern Recognition 2001 (CVPR 2001)*, volume 1, pages 511–518, 2001.

[21] R. Lienhart and J. Maydt. An extended set of haar-like features for rapid object detection. In *IEEE International Conference on Image Processing 2002 (ICIP 2002)*, volume 1, pages I–900 – I–903, 2002.

[22] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Proceedings of the 9th European Conference on Computer Vision 2006 (ECCV 2006)*, pages 404–417, 2006.

[23] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.

[24] Andrew H. Jazwinski. *Stochastic Processes and Filtering Theory*. Dover, Inc., 2007.

[25] S.J. Julier, J.K. Uhlmann, and H.F. Durrant-Whyte. A new approach for filtering nonlinear systems. In *American Control Conference, Proceedings of the 1995*, volume 3, pages 1628–1632 vol.3, 1995.

[26] Arnaud Doucet, Nando De Freitas, and Neil Gordon, editors. *Sequential Monte Carlo methods in practice*. "Springer", 2001.

[27] The OSGi Alliance. OSGi service platform core specification, release 5. http://www.osgi.org/Specifications, 2012.

[28] Hyuk Lim, Lu-Chuan Kung, Jennifer C Hou, and Haiyun Luo. Zero-configuration indoor localization over ieee 802.11 wireless infrastructure. *Wireless Networks*, 16(2):405–420, 2010.

[29] Michael G Wing, Aaron Eklund, and Loren D Kellogg. Consumer-grade global positioning system (gps) accuracy and reliability. *Journal of Forestry*, 103(4):169–173, 2005.

[30] Andras Csepinszky, François Fischer, and Maxime Flament. Intelligent vehicle systems and cooperative systems fots: Using eurofot operational experience in drive c2x. In *18th ITS World Congress*, 2011.

[31] O. Boyaci, A. Forte, S.A. Baset, and Henning Schulzrinne. vdelay: A tool to measure capture-to-display latency and frame rate. In *Multimedia, 2009. ISM '09. 11th IEEE International Symposium on*, pages 194–200, 2009.

[32] David L Mills. Internet time synchronization: the network time protocol. *Communications, IEEE Transactions on*, 39(10):1482–1493, 1991.