


# TUMTraf Event: Calibration and Fusion Resulting in a Dataset for Roadside Event-Based and RGB Cameras

Christian Creß\*  Walter Zimmer  Nils Purschke Bach Ngoc Doan Sven Kirchner  
Venkatnarayanan Lakshminarasimhan Leah Strand Alois C. Knoll 

**Abstract**—Event-based cameras are predestined for Intelligent Transportation Systems (ITS). They provide very high temporal resolution and dynamic range, which can eliminate motion blur and improve detection performance at night. However, event-based images lack color and texture compared to images from a conventional RGB camera. Considering that, data fusion between event-based and conventional cameras can combine the strengths of both modalities. For this purpose, extrinsic calibration is necessary. To the best of our knowledge, no targetless calibration between event-based and RGB cameras can handle multiple moving objects, nor does data fusion optimized for the domain of roadside ITS exist. Furthermore, synchronized event-based and RGB camera datasets considering roadside perspective are not yet published. To fill these research gaps, based on our previous work, we extended our targetless calibration approach with clustering methods to handle multiple moving objects. Furthermore, we developed an early fusion, simple late fusion, and a novel spatiotemporal late fusion method. Lastly, we published the TUMTraf Event Dataset, which contains more than 4,111 synchronized event-based and RGB images with 50,496 labeled 2D boxes. During our extensive experiments, we verified the effectiveness of our calibration method with multiple moving objects. Furthermore, compared to a single RGB camera, we increased the detection performance of up to +9% mAP in the day and up to +13% mAP during the challenging night with our presented event-based sensor fusion methods. The TUMTraf Event Dataset is available at <https://innovation-mobility.com/tumtraf-dataset>.

**Index Terms**—Event-Based Cameras, RGB Cameras, Sensor Fusion, Targetless Calibration, Multi-modal Dataset, Intelligent Transportation Systems

## I. INTRODUCTION

The principle of event-based cameras is the asynchronous recognition of changes in the brightness of each pixel. This technique results in a very high temporal resolution and a very high dynamic range [1], [2]. Therefore, an event-based camera is predestined for Intelligent Transportation Systems (ITS). Even in difficult visibility conditions, these systems require robust and accurate perception of traffic participants. Here, event-based cameras can achieve significant improvements, e.g., at night, with poor visibility or even fast-moving objects, leading to motion blur when using conventional cameras. However, the disadvantages of these novel sensors are the lack

All authors are with the Chair of Robotics, Artificial Intelligence and Real-time Systems, TUM School of Computation, Information and Technology, Technical University of Munich, Munich, Germany.  
E-mail: christian.cress@tum.de, knoll@in.tum.de.

\*Corresponding author.

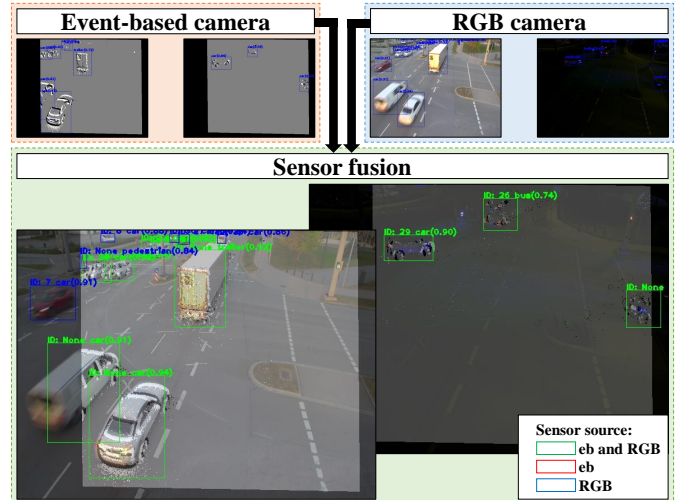


Fig. 1: This figure shows the sensor fusion between event-based and RGB cameras and its impact during a sunny day and a night in sleet. The blue bounding boxes in the event-based respectively RGB camera section represent detections without fusion. However, in the sensor fusion section, a green bounding box indicates that an object was detected by the event-based and the RGB camera. A blue bounding box shows detection exclusively by the RGB camera, and a red bounding box shows detection exclusively by the event-based camera (not available here). A unique track ID is assigned when objects are detected in several frames.

of color and texture information compared to conventional cameras and the fact that event-based cameras only detect moving objects. An optimized combination of roadside event-based and conventional RGB cameras offers advantages from both modalities. So far, ITS perception mainly relies on conventional cameras, Radars, and Lidars. Event-based cameras are yet to be widespread but are slowly being established in this area [3]. For this reason, an investigation into calibration, detection, and data fusion between roadside event-based and conventional cameras is necessary.

Detection and tracking with a stationary event-based camera mounted on roadside ITS, as far as we know, was first performed by [4]. Here, clustering and tracking methods achieved sufficient performance. In addition, [5] used a stationary event-based camera for pedestrian detection with a convolutional

neural network (CNN). For data fusion, the authors [6]–[9] carried out fusion on the feature level from a perspective with ego-motion. Furthermore, [9] performed early fusion, and [10] performed late fusion. Nevertheless, a fusion between event-based and RGB cameras is a developing field [9]; more knowledge about this topic in the area of stationary sensors in ITS is desirable. The authors [11]–[21] provided datasets from an ego-motion perspective. Unfortunately, data with ego-motion significantly differs from data from stationary cameras. Simulators [12], [22] could tackle this problem but suffer from the sim2real gap [9]. Another possibility is using pseudo-labels based on RGB camera detections [23]. This approach promises sufficient results, but an extrinsic calibration between event-based and RGB cameras is required. To calculate the intrinsic camera matrix, the authors [24]–[30] used classical checkerboards, which were moved in front of the camera, or used checkerboards, which emitted changes in brightness. However, calibration patterns are impractical on an ITS. Therefore, our previous work [31] presented a novel targetless extrinsic calibration method between event-based and RGB cameras. The approach produced adequate results but needed to be more robust in situations with multiple moving objects (e.g., several cars or shadows), which were not equally imaged by both cameras. These gaps are intended to be closed with this work.

To the best of our knowledge, there exists neither a targetless calibration approach that can handle multiple moving objects nor a data fusion between event-based and RGB cameras that takes the unique characteristics of a stationary camera setup during day and night in the domain of roadside ITS into account. Furthermore, there is still a lack of datasets in the mentioned domain.

For this reason, we improve in this work our previously presented targetless calibration approach between event-based and RGB cameras [31] to increase the practicability and the handling of multiple moving objects. Furthermore, to combine the advantages of both sensor modalities, we provide three fusion approaches between event-based and conventional cameras: Early Fusion (EF), Simple Late Fusion (SLF), and Spatiotemporal Late Fusion (STLF) based on SORT [32] tracking. Here, we demonstrate the effectiveness of our calibration and fusion methods with comprehensive experiments based on real data and comparisons with state-of-the-art datasets and fusion methods. For the experiments with sensor fusion, we analyzed the combination of event-based and RGB cameras during a sunny day and a night in the sleet on our ITS [33], see Figure 1. Lastly, we would like to share our novel TUMTraf Event Dataset. It contains synchronized event-based and RGB images, which show a complex road intersection with several traffic scenarios during the day and night. The dataset labels for training and validation are based on partially optimized pseudo-labels extracted from the YoloV7 [34]–[36] detector using an extrinsic calibration matrix obtained by our targetless calibration tool. The test dataset is carefully labeled and allows accurate ground truth data analysis.

In summary, the main contributions of this work are:

- Based on our previous work [31], an improved targetless calibration between event-based and RGB cameras,

which can handle multiple moving objects.

- The fusion methods Early Fusion (EF), Simple Late Fusion (SLF), and Spatiotemporal Late Fusion (STLF) between event-based and RGB cameras to profit from both sensor modalities’ advantages and reduce their limitations.
- Comprehensive experiments with our fusion algorithms based on real data and comparisons with other state-of-the-art datasets and methods.
- The novel TUMTraf Event Dataset, which contains spatiotemporal calibrated event-based and RGB images. The dataset can be used to show robust detection of traffic participants with an event-based camera during day and night in the domain of Intelligent Transportation Systems.

## II. RELATED WORK

First, we give an overview of event-based cameras and their usage in roadside ITS. Furthermore, we briefly summarize existing calibration algorithms for multi-sensor setups and state-of-the-art detection and fusion between event-based and conventional cameras.

### A. Event-based cameras in roadside ITS

Event-based cameras recognize changes in the brightness of each pixel asynchronously. As explained in our previous work [31], which refers to [2], each pixel responds independently to brightness changes in the continuous log brightness signal  $L(\mathbf{u}_k, t)$ . Here, an event  $e_k = (\mathbf{u}_k, t_k, p_k)$  is triggered at pixel  $\mathbf{u}_k = (x_k, y_k)^T$  at time  $t_k$  when the brightness change  $\Delta L(\mathbf{u}_k, t_k)$  since the last event at the same pixel  $\Delta t_k$  reaches a threshold  $C$ :

$$\Delta L(\mathbf{u}_k, t_k) = L(\mathbf{u}_k, t_k) - L(\mathbf{u}_k, t_k - \Delta t_k) = p_k C, \quad (1)$$

where  $C > 0$ . The event polarity is the sign of the brightness change  $p_k \in \{+1, -1\}$ . Visually, we can interpret an event as motion. As a result of this, event-based cameras have a very high temporal resolution and a dynamic range of 140 dB, far more than conventional cameras (60 dB) [2]. So, the perception system of an ITS could benefit from low energy consumption, low latency, and higher detection performance in challenging conditions (night vision, no motion blur from high-speed vehicles) [1]. Because of these advantages, event-based cameras are slowly being established in roadside ITS [3]. The authors of [4] presented the first and only detection and tracking approach we know, using a stationary event-based camera mounted on roadside infrastructure. The approach contains several clustering (e.g., DBSCAN [37]) and tracking methods (e.g., SORT [32]) for object detection. They achieved sufficient detection performance with more than 110 Hz frame rate. However, the authors noted a lack of datasets in the ITS domain. In this scope, we recognize research gaps in using state-of-the-art CNNs, analysis of performance in different lighting conditions (e.g., day or night), and the fusion with other sensor systems in the domain of roadside ITS.

TABLE I: This table lists popular event-based camera datasets. Here, we compare the general purpose, the perspective, the resolution of the event-based camera used, and the illumination scenarios of the datasets. Furthermore, in the case of 2D detection and lane extraction, we analyze the number of labeled frames, object classes, and labels.

General Properties						2D Detection Properties		
Name	Year	Purpose	Perspective	EB-Resolution	Day Night	# Lab. Frames	# Classes	# Labels
DDD17 [11]	2017	End-to-End Driving	Vehicle	346 × 260	<b>X X</b>			
DAVIS 240C [12]	2017	Depth & Pose	Mov. objects	240 × 180	X -			
N-CARS [13]	2018	2D Detection	Vehicle	304 × 240	X -*	48,000	2	24,029
MVSEC [14]	2018	Depth & Pose	Mov. objects	346 × 260	X -			
DET [15]	2019	Lane Extraction	Vehicle	<b>1280 × 800</b>	X -*	5,424	1	17,103
DDD20 [16]	2020	End-to-End Driving	Vehicle	346 × 260	<b>X X</b>			
1 MADD [17]	2020	2D Detection	Vehicle	1280 × 720	X -	<b>3,164,400</b>	3	<b>25,000,000</b>
ATIS ADD [18]	2020	2D Detection	Vehicle	304 × 240	X -*	<u>1,404,000</u>	2	255,781
DSEC-Det. [19]–[21]	2023	2D Detection	Vehicle	640 × 480	<b>X X</b>	<u>70,379</u>	<b>8</b>	<u>390,118</u>
TUMTraf Event (Ours)	2024	2D Detection	<b>Roadside</b>	640 × 480	<b>X X</b>	4,111	<u>7</u>	50,496

\*Information was not available.

## B. Calibration

Data fusion between event-based and conventional cameras requires an accurate calibration. We can generally distinguish between target-based (e.g., with a checkerboard pattern) and targetless methods. [24], [25] used a classical checkerboard, which was moved in front of the camera. [26] also calibrated an event-based camera with a checkerboard, whose lighting was changed using a flashlight to trigger events. Furthermore, [27], [28] apply a flashing LED grid pattern and [29], [30] apply a flashing screen with a shown checkerboard. The usage of these targets in the domain of roadside ITS is impracticable. Consequently, a targetless method is required.

In general, extrinsic calibration algorithms for multi-sensor systems are based on the principle of registering similarities. Due to thermal factors, wind gusts, or other uncontrolled movements, [38] performed a multi-camera autocalibration to improve the calibration accuracy during runtime. [39] calibrated targetless cameras, thermal cameras, and laser sensors, based on key points extraction in the images using SIFT [40] and point cloud registration between the modalities using ICP [41]. The extrinsic calibration approach between camera and Lidar of [42] used the assignment of the point cloud segmentation map and the semantic segmentation map of the camera image. Another calibration approach for camera and Lidar in an unstructured environment using Kalman Filter was developed by [43]. The authors [44] performed extrinsic targetless calibration between a stationary camera and Radar based on point cloud clustering with DBSCAN [37]. Each Radar object cluster is tracked first, then assigned with the camera detections. Similar to this, [45] also have developed calibration for camera and Radar using the Hungarian Method to find an association between the Radar clusters, generated via DBSCAN [37], and the camera bounding boxes, generated via YoloV3 [46]. Interestingly, [45] provided an automatically labeled Radar dataset based on the RGB detections.

In our previous work [31], we developed a targetless calibration method. In this work, we noticed that established image registration methods (e.g., SIFT [40]) cannot be applied to event-based images. Therefore, we extracted the edges of moving objects and registered them to each other. Our approach provided sufficient calibration accuracy if the

cameras captured the same object. The main limitation was the robustness against disturbances (e.g., shadow) or multiple moving objects not equally imaged by both cameras. In these cases, the algorithm was unable to produce accurate results. To the best of our knowledge, we are unaware of other targetless calibration methods for stationary event-based cameras.

## C. Detection

Object detection with event-based cameras can be realized using an unsupervised learning method (e.g., clustering), spiking neural networks (SNNs), or convolutional neural networks (CNNs). Here, the weights of the neural networks can be initialized with pre-trained knowledge from a different domain [1]. Based on these methods, robust object detection or several fusion approaches can be implemented. As already mentioned, [4] used several clustering methods (e.g., DBSCAN [37]) for object detection with event-based cameras. In Contrast, [5] applied a YoloV3 [46] object detector for pedestrian detection with a stationary event-based camera.

A primary factor for high-quality detections is a broad dataset. Therefore, [11]–[21] provided data from an event-based camera from an ego-motion perspective (e.g., recorded from a vehicle). Table I gives an overview of the datasets mentioned and shows their purposes and properties. Unfortunately, there are significant differences in the data between an event-based camera with ego-motion and a stationary-mounted event-based camera on a roadside ITS: In addition to the difference in the perspective, the representation of non-moving objects is logically not displayed in stationary event-based cameras. Although many datasets have been published, there is still a lack of labeled event-based camera datasets from a stationary roadside perspective [1], [5]. To tackle this problem, the simulators [12], [22] could create synthetic datasets. However, particularly for event-based cameras, [9] mentioned the existing sim2real gap. Imaginative intelligence, e.g., the Text-to-Video generator Sora [47], is a further method to create synthetic datasets. Here, an image similar to event-based camera output could be made with suitable prompts if the trained model depicts the spatial-temporal relationships and physical laws. In this case, scenario engineering has to ensure the safe and robust application of artificial intelligence for object detection with event-based cameras. For

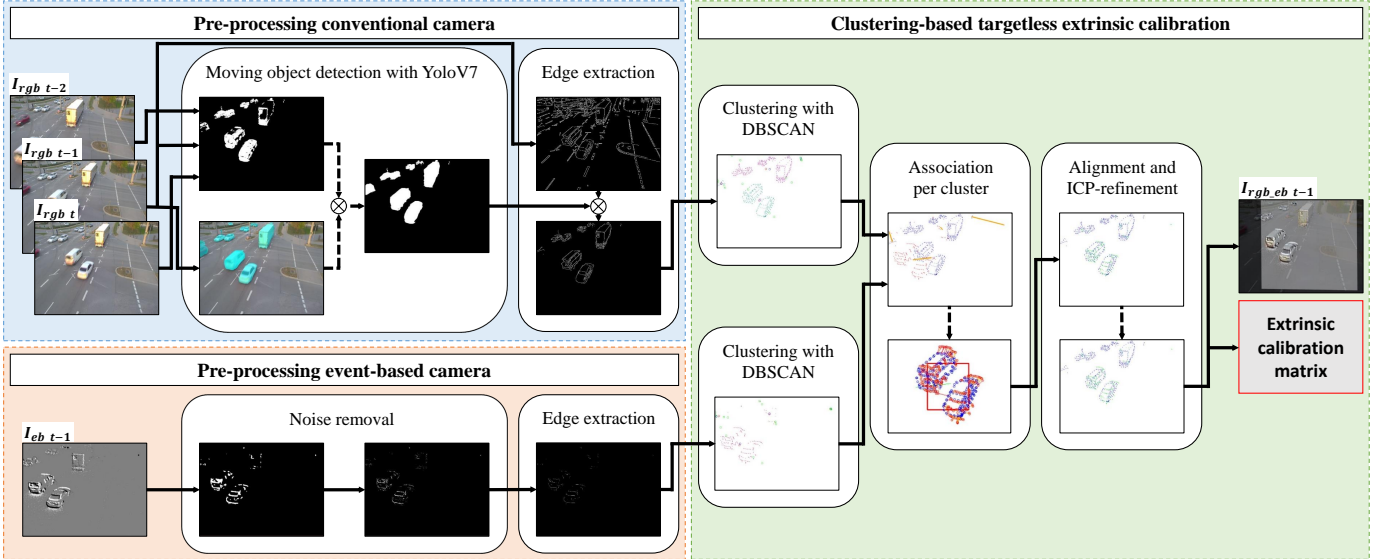


Fig. 2: The main components of our targetless extrinsic calibration algorithm are “pre-processing conventional camera,” “pre-processing event-based camera,” and “clustering-based targetless extrinsic calibration.” The event-based camera indicates moving image regions. However, we identify such areas in the RGB camera by analyzing the last three images. We extended our previous work [31] with DBSCAN [37] and can now handle multiple moving objects. The fundamental goal is to find associations per cluster pair to calculate a global transformation matrix. This approach allows us to calibrate in more complex traffic scenarios.

this purpose, the framework “scenario engineering-enabled foundation models in metaverse,” proposed by [48], could be used. Another way to get around the lack of labeled datasets is the approach of [23], which is generated in a multi-sensor setup using pseudo-labels. With this, the authors transformed the detections of an RGB camera into the domain of an event-based camera and used them as labels. This approach promises sufficient results for training CNNs with event-based images and is also used in our work. At this point, we would like to note that these CNN models are still black boxes and, therefore, don’t allow guarantees for certification. A trustworthy artificial intelligence is needed. The theoretical scenario engineering approach framework of the authors [48], [49] includes verification and validation and could solve these challenges in future work.

#### D. Fusion

In addition to the previously mentioned strengths of event-based cameras, there are limitations, such as the lack of color and texture information [6], [7]. Nevertheless, these systems can complement frame-based RGB cameras with intelligent data fusion [9]. However, compared to the fusion of Radar, camera, and Lidar, the fusion with event-based cameras is a relatively nascent field [9]. According to [50], we can distinguish between the levels “Data level (early) fusion,” “Feature level (middle) fusion,” and “Decision level (late) fusion.” On an early level, raw data is fused. This approach is useful when the data of the different modalities are comparable and compatible. Fusion on the feature level extracts features from each modality and combines them before being fed into a classifier. The output can be used in learning algorithms. In contrast, late fusion considers the detections of each sensor

modality and combines them to produce a final decision. This optimal assignment between the object detections can, e.g., be implemented using a modified Jonker-Volgenant algorithm [51], as in the late fusion for camera and Lidar of [52].

Several data fusion approaches for cameras have already been developed. First, we want to highlight the fusion approach IFCNN [53]. This approach uses a convolutional neural network to extract distinctive image features of several image modalities and fuse them by an appropriate fusion rule. Finally, the fused features are reconstructed by two convolutional layers to produce the informative fusion image. This fusion approach is particularly noteworthy because of its generalizability: In their experiments, the authors have achieved impressive results in fusing various types of images, such as multi-focus, infrared-visual, multi-modal medical, and multi-exposure images. The authors [6]–[9] also fused the separately extracted features of event-based and RGB images generated from a perspective with ego-motion. Here, [9] chose a voxel grid to represent event-based data and extracted their features. After applying the homography transformation from event-based to the RGB camera, the features were fused and fed as input to a RetinaNet [54] for object detection. The dataset used was also recorded from a moving vehicle. In addition, [9] experimented with image reconstruction using the method proposed in [55] with the result of high computation costs. They also performed an early fusion and detection by combining the RGB and event voxels. Furthermore, a late fusion between event-based and RGB cameras was developed by [10]. The authors used DBSCAN [37] clustering as a detector for the event-based camera and RetinaNet [54] for the RGB camera. Last but not least, the approach VistaGPT of [56] is based on large language models and could divide and recombine the complete perception pipeline, allowing a novel

hybrid form of all mentioned fusion levels. Nevertheless, there is still a lack of knowledge about the detection performance of early and late fusion between a stationary event-based and RGB camera in the domain of ITS with real data under adverse weather conditions.

### III. METHODOLOGY

In this section, we describe our targetless extrinsic calibration approach, which is an essential improvement of our previous work [31]. Furthermore, we present our method to generate a synchronized event-based and conventional RGB camera dataset, as well as the object detector that we used to train on our dataset. Lastly, we close this section with our developed Early Fusion, Simple Late Fusion (SLF), and novel Spatiotemporal Late Fusion (STLF) algorithms between event-based and RGB cameras optimized for stationary usage at a roadside ITS.

#### A. Targetless calibration

Figure 2 gives an overview of our extrinsic calibration approach based on multiple objects. As mentioned in our previous work, the image content of event-based cameras is indicated by motion in the scene. The optical flow represents the motion in images from conventional cameras. For calculating the extrinsic calibration matrix, this approach aims to accurately match the detected motion between event-based and RGB cameras to find image correspondences between both sensor modalities. The calibration algorithm can be divided into “pre-processing conventional camera,” “pre-processing event-based camera,” and “clustering-based targetless extrinsic calibration.” With this cluster analysis, we can tackle the main weakness of our previous work [31] that only one dynamic object can be in the field of view of all cameras. The other assumptions are still valid: We need a time-synchronized event-based and conventional camera setup for correct targetless calibration that recognizes the same objects from almost the same perspective. In addition, the objects must have a sufficient distance from the camera to be considered as a planar plane.

In the first step, as in our previous work [31], our algorithm accurately extracts the edges of moving objects in the event-based camera. This grayscale image contains the brightness changes accumulated over the last 5000  $\mu$ s. White areas indicate event polarity of +1, and black areas of -1. Gray areas show no motion. For simplification, we ignore the polarity of the events and, therefore, convert the image into a black-and-white image: Black defines static image areas and white dynamic image areas. In contrast to [31], we directly apply a dilation operation on the image with a kernel size of  $ksize = 3 \times 3$ . Then, a median filter with  $ksize = 3$  removes noise from the binary image. The faster an object moves, the larger the white area in the processed event-based image. As previously shown in [31], to enhance an edge image  $E \in \mathbb{R}^2$ , we apply efficient morphological hit-miss operations

using a combination of structuring elements (kernels  $K \in \mathbb{R}^3$ ) in vertical, horizontal and diagonal directions, as follows:

$$\begin{aligned} K_{verti} &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & 1 & 0 \end{bmatrix}, K_{horiz} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \\ K_{diag1} &= \begin{bmatrix} 0 & -1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}, K_{diag2} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \end{aligned} \quad (2)$$

Then, we combine the edge images based on the kernels mentioned, with  $E = E_{verti} + E_{horiz} + E_{diag1} + E_{diag2}$ .

In the second step, we detect the edges of multiple moving objects in the conventional camera. To enable more accurate motion detection, inspired by [57], unlike [31], we extract motion  $M_t \in \mathbb{R}^2$  with the last three grayscale images  $I_t \in \mathbb{R}^2$ ,  $I_{t-1}$ , and  $I_{t-2}$  and a binary threshold with  $T \in \mathbb{Z}$ , e.g.,  $T = 10$ :

$$\begin{aligned} D1_t &= |I_t - I_{t-1}|, & D2_t &= |I_{t-1} - I_{t-2}|, \\ T1_t &= \begin{cases} 255 & \text{if } D1_t > T \\ 0 & \text{otherwise} \end{cases}, & T2_t &= \begin{cases} 255 & \text{if } D2_t > T \\ 0 & \text{otherwise} \end{cases}, \\ M_t &= T1_t \wedge T2_t. \end{aligned} \quad (3)$$

Nevertheless, as in [31], we only consider motion caused by moving objects, not environmental influences, e.g., camera vibrations due to wind gusts. Therefore, inspired by [58], we also analyze the optical flow based on the methods Good Features To Track by J. Shi and Tomasi [59] and Lucas-Kanade optical flow in pyramids [60]: A flow vector  $v \in \mathbb{R}^2$  with a specific length  $l$  is assigned to camera motion if

$$v_l < (m + C), \quad (4)$$

with  $m$  as the median of the length of all optical flow vectors and  $C \in \mathbb{R}$  as a constant value, e.g.,  $C = 0.5$ . The other flow vectors indicate motion by moving objects. Furthermore, we also apply a KNN background subtractor [61] to receive the motion from the image sequence. To consider the edge extraction of the complete texture of a moving object, we use deep-learning-based instance segmentation provided by YoloV7 [34]–[36], pretrained on the MS Coco Dataset [62]. In contrast to our previous work [31], we don’t just consider the object containing the most movement. Instead, for calculation  $M_{yolo_t}$ , we consider all detected instance segmentation masks, where the ratio of the motion  $r_{motion}$  is greater than  $C_{motion} \in \mathbb{R}$ , e.g.  $C_{motion} = 0.2$ , and the ratio to the total image  $r_{total}$  is greater than  $C_{total} \in \mathbb{R}$ , e.g.  $C_{total} = 0.002$ , as follows:

$$r_{motion} = \frac{m_i}{d_i}, \quad r_{total} = \frac{d_i}{S} \quad \text{for } i = 0, \dots, n, \quad (5)$$

with the number of pixels  $m$  in the motion mask and the number of pixels  $d$  in the instance segmentation mask of each detected object  $i$ . Here,  $n$  is the total number of detected objects, and  $S$  is the total number of pixels in the image.

Then, we combine the motion mask  $M_t$ , described above, and the motion mask  $M_{yolo_t}$  with a logical OR operation.

TABLE II: The roadside perspective TUMTraf Event Dataset is designed for training an event-based and early fusion detector. “L-EB” labels are visible for the event-based camera, and “L-RGB” labels for the RGB camera. We generated the training and validation labels via pseudo-labeling. The test set was labeled manually to ensure accurate evaluation and was splitted into the subsets “Day”, “N-1” (“night with street lights on”), and “N-2” (“night with street lights off”). In total, the dataset consists of 7 classes and 50,496 labels. Unfortunately, we noticed a lack of pedestrians, bicyclists, and motorcycles in our recordings, particularly at night. For the sake of completeness, we have listed these classes anyway.

ID	Class	Train	Train	Val	Val	Test			Test		
		L-EB	L-RGB	L-EB	L-RGB	L-EB	N-1	N-2	L-EB	N-1	N-2
		Day		Day		Day	N-1	N-2	Day	N-1	N-2
0	Pedestrian	739	1,332	61	328	163	0	0	1,196	0	0
1	Bicycle	17	51	0	0	38	0	0	109	0	0
2	Car	<b>8,620</b>	<b>18,710</b>	<b>1,624</b>	<b>3,534</b>	<b>1,211</b>	<b>300</b>	<b>52</b>	<b>3,717</b>	<b>620</b>	<b>90</b>
3	Motorcycle	2	0	0	9	17	0	0	21	0	0
4	Bus	352	443	16	18	37	22	0	49	42	0
5	Truck	754	2,197	127	422	155	2	0	400	2	0
6	Trailer	698	1,097	112	399	225	0	0	366	0	0

With this procedure, we obtain motion, which includes moving traffic participants and background, e.g., shadows or blowing trees in the wind. To receive the edges of these moving objects, we first apply Canny edge detection [63] on the conventional camera image. Second, we combine it with the extracted motion mask via a bitwise AND operation.

At this point, similar edge images, including moving objects from the event-based and conventional camera, are available. To deal with multiple moving objects, we divide the edge images into several clusters using DBSCAN [37]. We want to emphasize the importance of dividing the event-based image into clusters, which has to be similar to the division of the conventional camera. After clustering, we determine the median centroids of each cluster. An optimal assignment between the clusters from event-based and conventional cameras can be found with these positions. For this purpose, the linear sum assignment problem will be solved with a modified Jonker-Volgenant algorithm [51].

Next, we search for an association inside each cluster pair.

For this, we must optimally align the event-based 2D point cloud with the conventional camera 2D point cloud. To be robust against outliers, we create an imaginary rectangle, see the red rectangle in Figure 2, between the 13<sup>th</sup> and 87<sup>th</sup> percentile for each event-based respectively conventional camera cluster. With these two rectangles  $r_{eb}$  and  $r_{rgb}$  for each cluster pair, we can, similar to our previous work [31], find a suitable scaling  $s \in \mathbb{R}^2$  and displacement  $t \in \mathbb{R}^2$  to transform each cluster of the event-based camera with  $T_{coarse} \in \mathbb{R}^3$ :

$$\begin{aligned} r_{rgb_w} &= r_{rgb_{x_{0.87}}} - r_{rgb_{x_{0.13}}}, \\ r_{rgb_h} &= r_{rgb_{y_{0.87}}} - r_{rgb_{y_{0.13}}}, \\ r_{eb_w} &= r_{eb_{x_{0.87}}} - r_{eb_{x_{0.13}}}, \\ r_{eb_h} &= r_{eb_{y_{0.87}}} - r_{eb_{y_{0.13}}}, \end{aligned}$$

$$s_x = \frac{r_{rgb_w}}{r_{eb_w}}, \quad s_y = \frac{r_{rgb_h}}{r_{eb_h}},$$

$$r_{eb_{scaled_x}} = s_x \cdot r_{eb_{x_{0.13}}}, \quad (6)$$

$$r_{eb_{scaled_y}} = s_y \cdot r_{eb_{y_{0.13}}}$$

$$t_x = (-1 \cdot r_{eb_{scaled_x}}) + r_{rgb_{x_{0.13}}},$$

$$t_y = (-1 \cdot r_{eb_{scaled_y}}) + r_{rgb_{y_{0.13}}},$$

$$\Rightarrow T_{coarse} = \begin{bmatrix} s_x & 0 & t_x \\ 0 & s_y & t_y \\ 0 & 0 & 1 \end{bmatrix}.$$

After translation and scaling, an optimal point-to-point assignment with the modified Jonker-Volgenant algorithm [51] inside of each cluster can be found. To filter outliers from the point assignments, we calculate the length of each point assignment  $i$  and determine the length  $l$  of the 70<sup>th</sup> percentile. If the assignment length  $l$  is greater than the length  $l$  of the 70<sup>th</sup> percentile, we calculate the extension factor  $f \in \mathbb{R}$ . If the factor  $f$  is greater than a threshold  $C \in \mathbb{R}$ , e.g.,  $C = 0.5$ , the point-to-point assignment  $i$  is defined as an outlier:

$$f_i = \frac{l_i - l_{p=0.70}}{l_i}, \quad f_i > C. \quad (7)$$

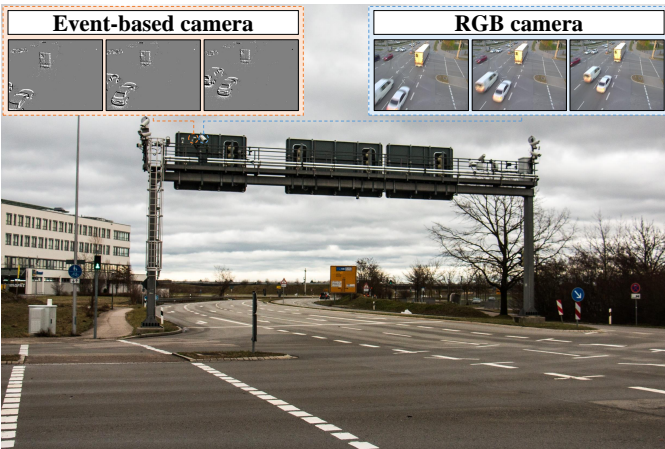


Fig. 3: We recorded the TUMTraf Event Dataset at this intersection in Garching near Munich. Besides event-based and RGB cameras, the gantry contains numerous other sensors, e.g., Lidars, which are the basis for the TUMTraf Dataset family.

After filtering the outliers, we consider only cluster pairs, where the minimum number of assignments  $M \in \mathbb{Z}$ , e.g.,  $M = 1$ , is achieved. Based on the remaining point-to-point assignments between event-based and conventional cameras, which we determined for each cluster pair, we calculate the coarse alignment as optimal affine transformation using RANSAC [64]. In the last step, we refine our coarse estimation between the point clouds of event-based and conventional cameras using point-to-point ICP [41].

### B. Dataset

According to [5] and Table I, a lack of event-based camera datasets with a stationary roadside perspective exists. Therefore, we create our TUMTraf Event Dataset, which includes spatiotemporal synchronized frame triplets of event-based, RGB, and combined RGB-event-based images, see Table II. For this, we first record the raw data with our ITS from the Providentia++ project [65], [66], which contains event-based and RGB cameras. As illustrated in Figure 3, the sensors are set up on a gantry at a height of 7 m located at an intersection in Garching near Munich, Germany. This perspective gives our dataset a unique bird’s eye view of the traffic, which minimizes the number of occlusions. The image pairs are recorded time synchronized. The spatial synchronization is achieved using our targetless calibration approach. With it, we create the combined RGB-event-based images with simple blending, see Subsection III-C. The specifications of the sensors used are as follows:

- **RGB camera:** Basler ace acA1920-50gc,  $1920 \times 1200$ , Sony IMX174, global shutter, color, GigE, with 8 mm lens.
- **Event-based camera:** Imago VisionCam EB,  $640 \times 480$ ,  $> 120$  dB dynamic range, 30 000 000 events/s, with 8 mm lens.

To receive accurate detections from the RGB camera, we train the YoloV7 object detector [34]–[36] on the nuImages Dataset [67], which counts 93,000 images, including rain, snow, and night scenarios from the ego perspective of a vehicle [67]. To consider the roadside perspective from a stationary camera on a gantry bridge, we perform transfer learning with the 2D annotations of our TUMTraf Dataset family [68], [69], which also includes snow and night scenarios. With this robust object detector for the conventional camera and the previously calculated extrinsic calibration matrix, we generate pseudo-labels, similar to [23], using the confidence threshold  $C = 0.80$ . This way, we can obtain any number of accurate pseudo-labels, enabling robust object detection with an event-based camera and early fusion.

We provide two categories of labels: The category “L-EB” includes only the labels of moving objects and can be used for the event-based image detector. Here, we automatically analyze each detected object with the optical flow and assign a motion attribute. The other category, “L-RGB,” consists of all objects that can, in principle, be recognized by the RGB camera, including moving and non-moving objects.

To enhance the dataset’s quality, we roughly filter the training and validation set by excluding frame triplets where

the pseudo-labels are obviously incorrect. This procedure results in an optimized dataset with 2,538 frame triplets, including synchronized event-based, RGB, and RGB-event-based combined frames for the training set, 580 frame triplets for the validation set, and 993 frames for the test set. True to the motto “Train during the day, detect at night,” we intentionally select only images from the scenario “Day” for the training and validation set. This choice enables us to achieve the best possible quality when generating the pseudo-labels. Image triples from the night scenarios are only used in the test set. This procedure is possible since the event-based camera has the same image content day and night. We also performed meticulous manual fine-tuning of each label in the test set to enable an accurate evaluation. The labels are available in OpenLABEL format [70].

### C. Detection and fusion

This section describes our detection and fusion methods between event-based and conventional cameras. In particular, our methods of Early Fusion (EF), Simple Late Fusion (SLF), and novel Spatiotemporal Late Fusion (STLF) combine the strengths of the two sensor modalities mentioned above. Figure 5 gives an overview of the processing pipeline for our multi-modal image fusion.

As a pre-processing step for the RGB camera, we calculate a motion mask using the grayscale images  $I_t$  and  $I_{t-1}$ . Then, we can determine the per-element absolute difference between both images and we get the motion mask  $M_t$  by applying a binary threshold function with threshold  $T$ :

$$D_t = |I_t - I_{t-1}|, \quad (8)$$

$$M_t = \begin{cases} 255 & \text{if } D_t > T \\ 0 & \text{otherwise} \end{cases}$$

Similar to our calibration approach, we also use a KNN background subtractor [61] for refinement. The calculated motion mask allows the fusion component to distinguish between static and moving objects.

Event-based cameras accurately recognize very rapid changes in brightness. If street lights are on, the accumulated grayscale image shows flickering. This phenomenon is due to the lamps operating with 50 Hz alternating current or pulse

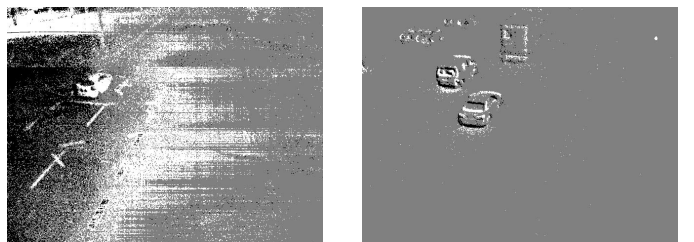


Fig. 4: Street lights cause significant noise at night, which must be eliminated with an anti-flickering filter. This phenomenon is due to the lamps operating with 50 Hz alternating current or pulse width modulation.

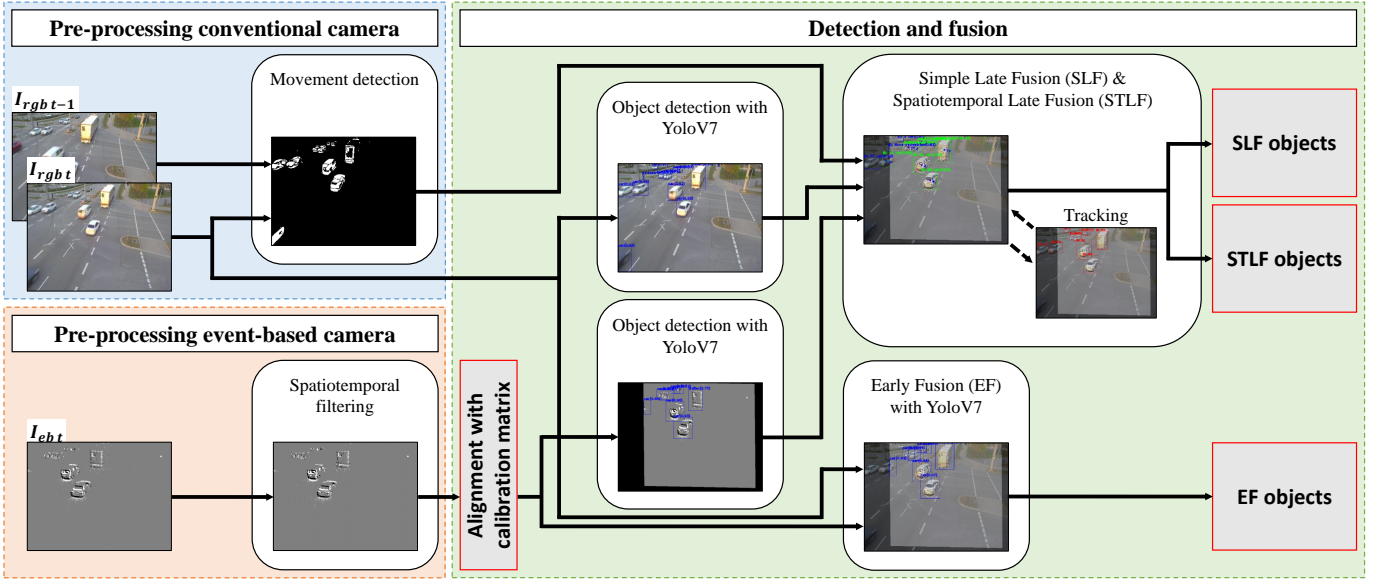


Fig. 5: This figure illustrates the processing pipeline of Early Fusion, Simple Late Fusion, and Spatiotemporal Late Fusion. Early Fusion uses the raw images from the RGB and event-based cameras. On the other hand, the late fusion methods operate on the detections based on the individual images and the motion mask of the RGB camera. In addition, Spatiotemporal Late Fusion utilizes tracking information of each object for its fusion decision.

width modulation. For this reason, the flickering is removed by the event-based camera driver, see Figure 4. Then, inspired by [71], a spatiotemporal filter is applied as a preprocessing step for data fusion. Here, for each event  $e_k = (\mathbf{u}_k, t_k, p_k)$  at pixel position  $\mathbf{u}_k = (x_k, y_k)^T$  at time  $t_k$ , we investigate the spatial-temporal neighborhood  $N = (N_x, N_y, N_t)$ , where

$$\begin{aligned} N_x &= [x_k - r_x, x_k + r_x], \\ N_y &= [y_k - r_y, y_k + r_y], \\ N_t &= [t_k - r_t, t_k], \end{aligned} \quad (9)$$

and  $r_x, r_y, r_t$  are the sizes of the neighborhood. The amount of events in the neighborhood  $N$  is defined as noise if the total number of events in the neighborhood does not achieve the threshold  $E \in \mathbb{Z}$ , e.g.  $E = 30$ . This efficient noise suppression ensures that only moving objects are in the image of the event-based camera.

The next step after pre-processing the event-based and conventional camera is the detection and fusion. Here, we develop the methods of Early Fusion, Simple Late Fusion, and a novel Spatiotemporal Late Fusion. For Early Fusion, we simply blend the event-based image  $I_{eb}$  and the RGB image  $I_{rgb}$  with

$$I_{eb\_rgb} = (1 - \alpha)I_{eb} + \alpha I_{rgb}, \quad (10)$$

and  $\alpha \in \mathbb{R}$ , e.g.,  $\alpha = 0.5$ . Then, we train with the label categories “L-EB” and “L-RGB” the YoloV7 detector [34]–[36] on the fused images  $I_{eb\_rgb}$ .

The YoloV7 detector [34]–[36] for the RGB camera is applied to the RGB images for the SLF and STLF. Analogous to this, the detector for the event-based camera, trained with “L-EB,” is applied to the aligned event-based images. In the second step, we use the previously calculated motion mask  $M_t$

and, thus, we find an optimal assignment using the modified Jonker-Volgenant algorithm [51] between objects detected by the event-based camera and moving objects detected by the RGB camera. If the Euclidean distance between the objects of each pair is greater than a threshold  $L \in \mathbb{R}$ , e.g.,  $L = 50.0$ , we reject the assignment. Otherwise, we create a fused object  $O_f$  and declare it as the output object of the fusion component. In principle, the RGB camera provides more texture information and is more precise in determining the object class. Consequently, the fused object receives the following properties with the weight  $\alpha \in \mathbb{R}$ , e.g.,  $\alpha = 0.4$ :

$$\begin{aligned} O_{f\_class} &= O_{rgb\_class}, \\ O_{f\_position} &= (1 - \alpha)O_{rgb\_position} + \alpha O_{eb\_position}, \\ O_{f\_size} &= (1 - \alpha)O_{rgb\_size} + \alpha O_{eb\_size}. \end{aligned} \quad (11)$$

In the third step, we take all fused and unfused objects of the event-based and RGB camera as output objects for the Simple Late Fusion so that moving and non-moving objects are considered. Here, we noticed that in difficult visibility conditions, e.g., night, the false positive rate for the RGB camera detector increases noticeably, even with confidence thresholds  $C = 0.70$ . Therefore, the YoloV7 detector [34]–[36] for the RGB images has to operate with a relatively high confidence threshold.

On the other hand, due to the noise-free grayscale mask, the detector of the event-based camera produces significantly fewer to no false positives, even in night images with confidence threshold  $C = 0.30$ , see Figure 8. For this reason, we develop a novel Spatiotemporal Late Fusion, where we track each fused object and potential output object with SORT [32] and assign a unique tracking ID. So, we can identify these objects in multiple frames over time. If an object was previously detected by the event-based camera or by a



TABLE III: We measured per frame the accuracy of our calculated extrinsic calibration and the manually created ground truth using the reprojection error in pixels. The sequences 1–3 are from our previous work [31]. Sequence 4 is part of the TUMTraf Event Dataset and contains numerous moving objects. Compared to our previous work, we achieved similar accuracy in all sequences, including the complex traffic scenarios.

#	Rep. Error (Rep. Error GT)			
	Sequence 1	Sequence 2	Sequence 3	Sequence 4
1	10.16 ( <b>1.15</b> )	8.69 (3.30)	<b>4.76</b> (1.55)	17.41 ( <u>1.62</u> )
2	10.76 (1.97)	11.09 ( <u>2.09</u> )	<u>6.42</u> (1.42)	15.60 ( <b>1.36</b> )
3	<b>3.37</b> (1.65)	14.49 ( <b>1.73</b> )	8.20 (1.99)	7.79 (1.90)
4	13.62 (2.76)	<b>8.67</b> (5.16)	6.69 ( <b>1.11</b> )	<b>6.72</b> (2.54)
5	6.05 ( <u>1.57</u> )	13.51 (2.14)	12.49 (1.84)	<u>7.70</u> (2.84)
6	<u>5.41</u> (1.79)	12.69 (2.17)	7.85 (1.69)	8.18 (3.40)

combination of event-based and RGB cameras, we classify this object as trustworthy. Subsequently, only objects from the RGB camera with a confidence threshold, e.g., greater than  $C = 0.77$ , or trustworthy objects are considered by STLF as output objects. This method allows us to significantly reduce the number of false positives.

#### IV. EVALUATION

In this section, we present the results of our improved targetless extrinsic calibration method based on our previous work [31]. In addition, we perform an ablation study with the DBSCAN [37] algorithm to discuss suitable parameters for a successful targetless calibration. Furthermore, we evaluate the performance of the event-based object detector based on our TUMTraf Event Dataset. Here, we compare our results (EB) to an event-based object detector in the use case of roadside event-based cameras, which we trained on the famous DSEC-Detection Dataset [19]–[21] (EB-DSEC). Lastly, we analyze in several traffic scenarios the strengths and limitations of our presented fusion approaches: early fusion based on “L-EB” labels (EF-1), early fusion based on “L-RGB” labels (EF-2), Simple Late Fusion (SLF), and Spatiotemporal Late Fusion (STLF). To classify these results, we carry out comparisons with the image fusion framework based on convolutional neural network (IFCNN) [53], and an early fusion based on the DSEC-Detection Dataset [19]–[21] (EF-DSEC). As a further ablation study, we also examined the “Confidence Threshold” hyperparameter for the method STLF.

##### A. Extrinsic Calibration

The main advantage of our improved targetless calibration between event-based and RGB cameras is the ability to handle multiple moving objects. We calculated the reprojection error based on ground truth data to evaluate the accuracy, which we manually created with the same tool as in our previous work [31]. For comparability to our previous work, we used the identical test sequences 1–3: sequences 1 and 3 contain a single moving car, and sequence 2 includes a crossing van with very slow oncoming traffic. We recorded these sequences with the same cameras as in the TUMTraf Event

Dataset; however, we used a 16 mm lens in our previous work. Furthermore, Sequence 4, which is part of the test set in the TUMTraf Event Dataset, includes bi-directional fast-moving traffic participants with different vehicle classes. We want to mention that our previous approach, fortunately, did not detect the slow movement of oncoming traffic in most frames in Sequence 2. Once multiple motions were detected, our previous approach could no longer determine a meaningful transformation matrix due to its coarse alignment procedure.

Table III shows the reprojection error of our extrinsic calibration and the error of the ground truth data in sequences 1–4. The results are roughly comparable to our previous work: We achieved in sequences 1 and 3 with a single moving vehicle a reprojection error of up to 3.37 px. In the more challenging Sequence 4, which contains several small, independently moving objects, we achieved a reprojection error of up to 6.72 px. Figure 6 shows the qualitative effectiveness of our targetless calibration, even with independently moving objects in complex traffic scenarios. Therefore, our improvement increases the flexibility of our previous work [31].

The DBSCAN [37] algorithm finds clusters based on the spatial density of a dataset and is essential for a successful targetless calibration. Its input parameters are the maximum allowed distance  $\epsilon$  between two data samples of a cluster and the minimum number of necessary data samples  $s_{min}$  to create a cluster. For our use case, the defined clusters from the event-based camera must correspond to those from the RGB camera. DBSCAN [37] is ideal for this task because we don’t have to specify the total number of clusters, and second, the algorithm is quite robust against outliers. An RGB camera offers significantly more texture information depending on the object than an event-based camera. Therefore, the generated clusters can differ significantly between both sensor modalities, even with the same parameters  $\epsilon$  and  $s_{min}$ . Here, we noticed that parameter  $s_{min}$  helps classify disturbing artifacts from the edge image (e.g., shadows) as noise, thus excluding them. In addition, parameter  $\epsilon$  should be set depending on the area the moving object occupies in the image. If there is only one moving object, the parameter can also be set as high as possible. Figure 7 shows three examples where we performed DBSCAN [37] in each example with two settings  $S_1$  and  $S_2$ . We defined the settings as follows:

$$S_1 = \{\epsilon^{eb} = 150, s_{min}^{eb} = 2, \epsilon^{rgb} = 150, s_{min}^{rgb} = 2\}, \quad (12)$$

$$S_2 = \{\epsilon^{eb} = 70, s_{min}^{eb} = 2, \epsilon^{rgb} = 40, s_{min}^{rgb} = 2\}. \quad (13)$$

As seen in examples 1–3, we achieved convincing calibration results if the clusters of the event-based and RGB cameras correspond. It is crucial to ensure no significant deviations, as in Example 2,  $S_2$ : Here, the DBSCAN [37] clustering divides the vehicle in the RGB image into two clusters, but the clustering in the event-based image represents the vehicle in only one cluster. As the targetless calibration algorithm scales the associated clusters to a uniform size, meaningful cluster matching is impossible in those cases.

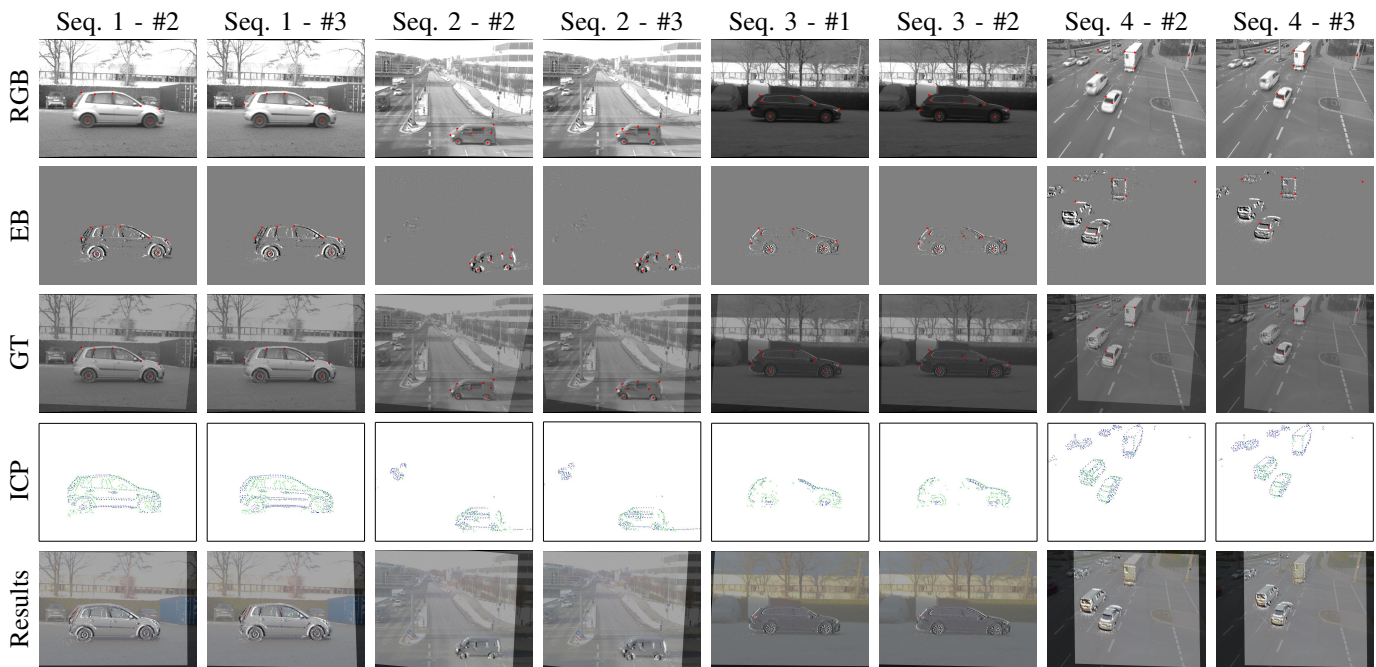


Fig. 6: The main goal of targetless extrinsic calibration is the association of features, e.g., edges, in both modalities. By manually selecting key points, we marked common points in the images from the event-based (EB) and RGB cameras and thus created a ground truth (GT). As can be seen, our improved targetless calibration is not only convincing for single-moving objects, but the approach can also carry out valid camera calibration in complex traffic scenarios with multiple-moving objects.

TABLE IV: Since fusion and detection are calculated for every frame on an ITS, the runtime performance is highly relevant. Preprocessing, essentially the extrinsic and intrinsic calibration of the camera images, requires the most computing time.

Component	Runtime
Preprocessing images	101 ms
Motion calculation	10 ms
CNN YoloV7 for RGB	17 ms
CNN YoloV7 for eb	15 ms
CNN YoloV7 for EF	15 ms
STLF / SLF	1.6 ms
<b>Total</b>	<b>159.6 ms</b>

All in all, we used the DBSCAN [37] setting  $S_2$  to calibrate the event-based and RGB images from the TUMTraf Event Dataset. Since the result of Sequence 4, frame #3, with a reprojection error of 7.79 px, provides the subjectively best extrinsic calibration result, all further data fusion experiments were carried out with this calculated transformation. The intrinsic parameters and the distortion models of the event-based and RGB cameras have been calibrated beforehand.

### B. Runtime analysis

Multi-modal sensor fusion ideally achieves more robust detection results by combining two sensor systems' strengths while eliminating their weaknesses. Since the fusion and detection is potentially calculated for every frame on an ITS, we would first like to examine the runtime performance. Table IV shows the runtime of the detection and fusion processing pipeline. Early Fusion requires 15 ms, while Simple Late

Fusion respectively Spatiotemporal Late Fusion requires, in a sequential view, 43.6 ms per frame. Interestingly, the most runtime requires pre-processing with 101 ms, which mainly includes the event-based camera filtering method and the application of the intrinsic and extrinsic calibration matrices. In future work, GPU parallelization could significantly accelerate the pre-processing overhead and the sequential processing in the late fusion methods. The measurements were made on an Intel Core i9-9900KF CPU, NVIDIA GeForce RTX 2080 SUPER - 8GB VRAM, and 32 GB RAM.

### C. Detection and fusion

The detector for RGB images serves as input for the late fusion methods, generates pseudo-labels for the training of the event-based camera detector, and represents a baseline for the fusion methods. As described in Subsection III-B, we first trained the YoloV7 CNN [34]–[36] on the nuImages Dataset [67]. Since the TUMTraf Dataset family considers roadside perspective, we finetuned the CNN to our TUMTraf Highway [68] and TUMTraf Intersection Dataset [69] using transfer learning. So, we significantly increased the detection performance from 0.36 mAP to 0.85 mAP on our combined TUMTraf test set.

Based on the RGB detector mentioned above and the extrinsic calibration, we generated the TUMTraf Event Dataset with pseudo-labels. We trained several object detectors for early and late fusion. The training with “L-EB” labels on event-based frames creates a robust detector for the event-based camera (EB), which we used for our late fusion approaches SLF and STLF. However, we trained the early fusion methods EF-1

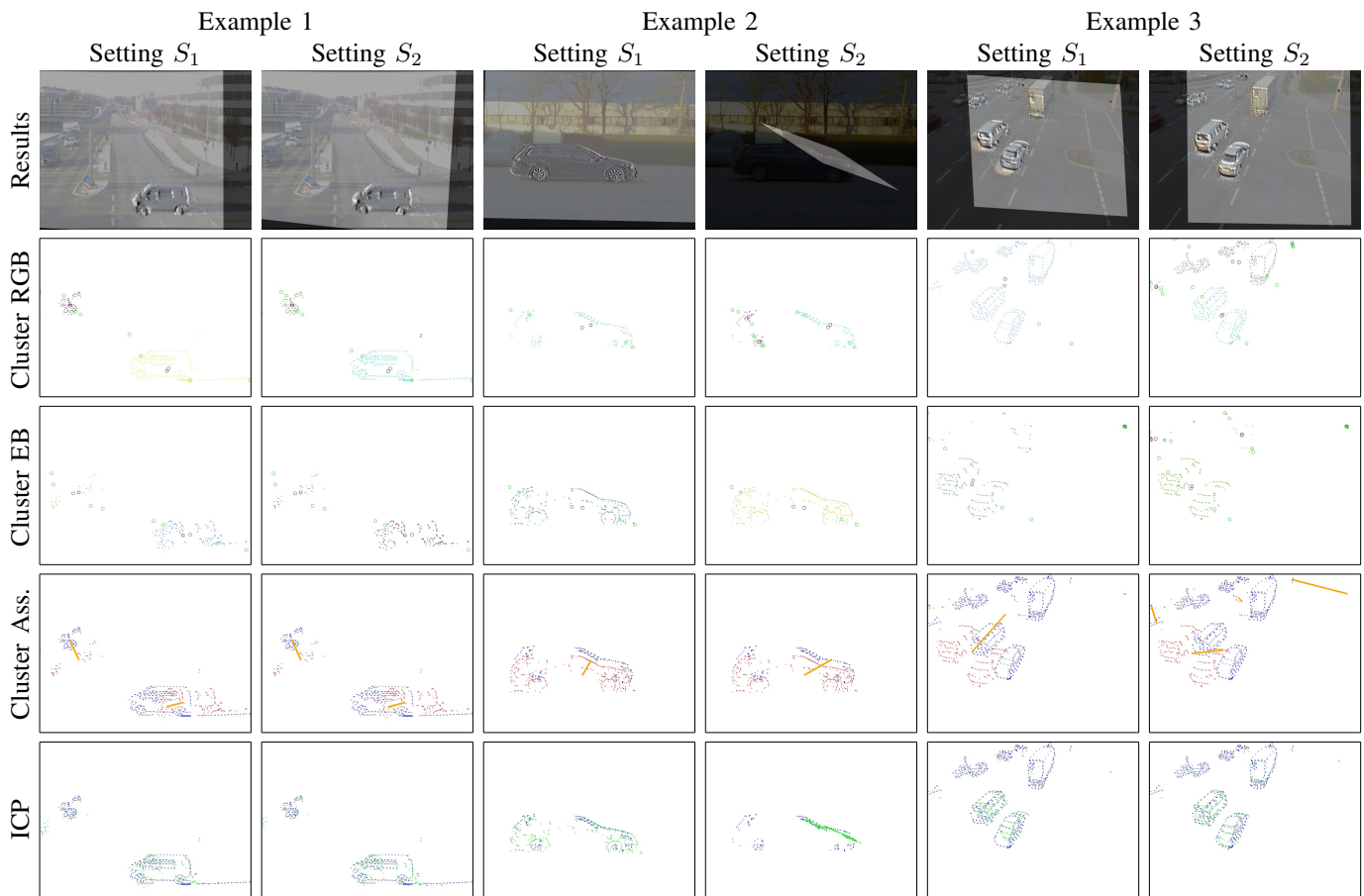


Fig. 7: The most influential in our targetless calibration are the DBSCAN [37] parameters of the maximum allowed distance  $\epsilon$  between two samples within a cluster and the minimum number of samples  $s_{min}$ , which are necessary to define a cluster. Our use case is to generate almost equal clusters in both sensor modalities. To investigate these parameters, we apply the settings  $S_1$  and  $S_2$  to three examples. This figure shows the final results, the emerging clusters, the cluster assignments, and the ICP output. High values for  $\epsilon$  allow a correct assignment for a single object, see  $S_1$  in Example 2. However, smaller values allow a more fine-grained and accurate assignment. Furthermore, the value  $s_{min}$  can successfully suppress noise during clustering.

and EF-2 on the combined RGB-event-based images. Here, we used the labels “L-EB” for EF-1 and “L-RGB” for EF-2. We want to emphasize that our training is based on the optimized variant of our training and validation set. Furthermore, we perform the evaluation on the carefully manually corrected test set. To measure the results, we used the toolkits of [35] and [72].

First, we analyze the performance of our event-based object detector (EB) with “L-EB” as ground truth. The quantitative results are in Table V, and the qualitative results are in Figure 8. In scenario “Day,” we achieved satisfactory results with the large object classes, e.g., car (0.44 AP) or bus (0.92 AP). These classes contain sufficient texture information and allow adequate detection. However, we recognized worse performance with optically small objects that don’t include enough features, e.g., pedestrians or bicycles. Another reason may be the low occurrence of these classes in the training set. Scenario “N-1” is a traffic scene at night with street lights on with buses and cars. The event-based detector still delivered satisfactory results: We could detect cars with 0.64 AP. Even in scenario “N-2,” a night with street lights off, we could detect cars with

0.33 AP and a precision of 0.88. At this point, the advantages of the event-based camera stand out clearly: 1.) The high dynamic range of the event-based camera allows the detection of objects under extreme illumination conditions. 2.) Since the stationary event-based camera provides a clear image mask, false positives in the background area are significantly reduced. Nevertheless, the recall of the class car compared to the “Day” scenario dropped significantly from 0.61 to 0.33 in absolute darkness. A possible cause could be disturbing artifacts, e.g., light beams from headlights, see Figure 8. These artifacts were mostly, but not completely, filtered with the spatiotemporal filtering by [71]. A data fusion between an event-based and RGB camera could combine the advantages of both modalities and thus reduce the disadvantage of a lower recall caused by less texture information.

For comparison, we additionally trained an event-based detector with the DSEC-Detection Dataset [19]–[21] (EB-DSEC). For this purpose, we rendered the event-based data as a grayscale image to have the same color code for event-based data as in the TUMTraf Event Dataset. Interestingly, EB-DSEC cannot reliably detect objects in all three scenarios.

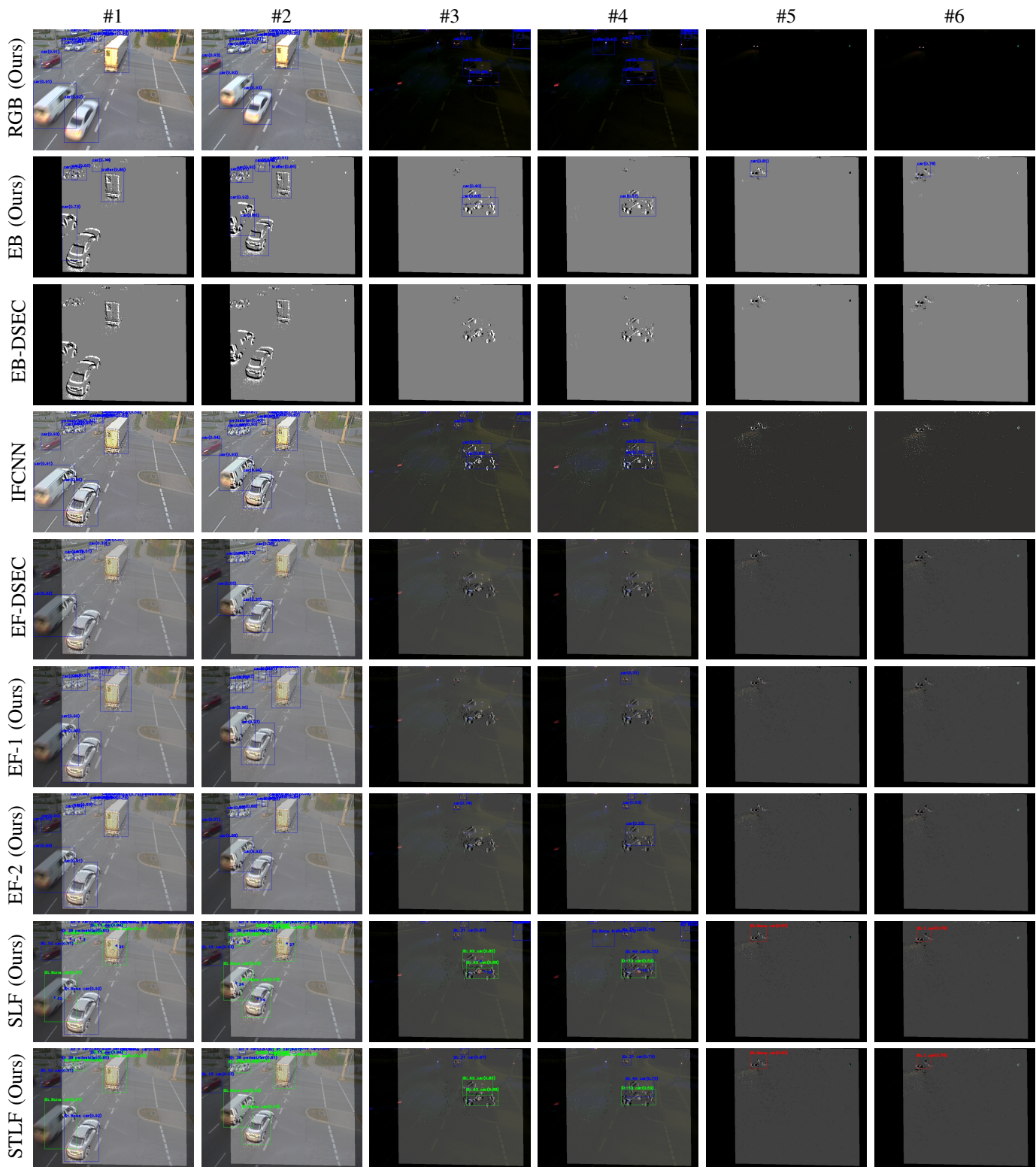


Fig. 8: This figure compares the results of the RGB and EB detectors and our fusion methods EF-1 and EF-2, SLF, and STLF with EB-DSEC and EF-DSEC, based on the DSEC-Detection Dataset [19]–[21] and the IFCNN [53] fusion. The experiments were on the “Day” (#1–#2), “night with street lights on” (#3–#4), and “night with street lights off” (#5–#6). Blue boxes in RGB, EB, IFCN [53], and EF are detections. In SLF and STLF, green boxes show detections by an event-based and RGB camera; a blue box indicates detection only by the RGB, and a red box only by the event-based camera. STLF successfully filtered false positives caused by the RGB camera in frames #3 and #4. The advantages of the event-based camera in darkness are visible. Thresholds: Confidence = 0.3; IoU = 0.45.

TABLE V: This table shows the average precision, precision, and recall (AP, P, R) of the event-based detector based on our TUMTraf Event Dataset (EB). We conducted experiments during the “Day,” at night with street lights on (“N-1”), and at night with street lights off (“N-2”). Here, we used the “L-EB” labels, representing moving objects and, therefore, being visible to the event-based camera. We achieved 0.26 mAP during the “Day” and up to 0.54 mAP at night. Furthermore, we compared our detector to a trained detector based on the DSEC-Detection Dataset [19]–[21] (EB-DSEC). Due to the ego perspective of EB-DSEC, object detection on our test set from the bird’s eye perspective is impossible. This result underlines the relevance of our TUMTraf Event Dataset. Thresholds: Confidence = 0.3; IoU = 0.45.

Testset		Class	EB-DSEC			EB		
Day	Full	Pedestrian	0.00	0.00	0.00	<b>0.01</b>	0.10	0.08
		Bicycle	0.00	0.00	0.00	0.00	0.00	0.00
		Car	<u>0.04</u>	0.80	0.04	<b>0.44</b>	0.61	0.61
		Motorcycle	0.00	0.00	0.00	0.00	0.00	0.00
		Bus	0.00	0.00	0.00	<b>0.92</b>	0.73	0.95
		Truck	0.00	0.00	0.00	<b>0.10</b>	0.28	0.13
		Trailer	0.00	0.00	0.00	<b>0.35</b>	0.83	0.37
	∅ Total	<u>0.01</u>	0.11	0.01	<b>0.26</b>	0.36	0.31	
N-1	Full	Car	<u>0.01</u>	0.22	0.01	<b>0.64</b>	0.83	0.74
		Bus	0.00	0.00	0.00	<b>0.44</b>	0.91	0.46
		∅ Total	<u>0.01</u>	0.11	0.01	<b>0.54</b>	0.87	0.60
N-2	Full	Car	0.00	0.00	0.00	<b>0.33</b>	0.88	0.33
		∅ Total	0.00	0.00	0.00	<b>0.33</b>	0.88	0.33

We suspect the domain gap of perspective: The training data consider just the ego perspective of a vehicle. The arrangement of the edges differs significantly compared to the roadside perspective from a height of 7 m; meaningful recognition seems impossible. This result underlines the relevance and the need for the TUMTraf Event Dataset.

In the next step, we evaluated our early and late fusion methods EF-1 and EF-2, SLF, and STLF. We compared our results with the RGB detector with two confidence thresholds as a baseline and the fusion methods IFCNN [53] and EF-DSEC, where we trained an early-fusion based on the DSEC-Detection Dataset [19]–[21]. To prepare the IFCNN [53] evaluation, we fused our complete TUMTraf Event Dataset with IFCNN [53] and trained and tested a YoloV7 [34]–[36] detector. Here, we used as ground truth “L-RGB” labels on our carefully manually corrected test set, which includes the scenarios “Day,” “N-1” (“night with street lights on”), and “N-2” (“night with street lights off”). We also examined various traffic scenarios. For this purpose, if available, we have extracted from the three illumination scenarios the subcategories “Standing,” “Vertical,” and “Horizontal,” which contain a specific dominant traffic flow. The quantitative results are in Table VI, and the qualitative results are in Figure 8.

In the scenario “Day,” the RGB detector achieved an mAP of 0.69, particularly with the object classes car an AP of 0.94 and bus the highest AP of 0.98. Nevertheless, we also recognized worse performance with optically small objects, e.g., pedestrians (0.64 AP), motorcycles (0.37 AP), or bicycles (0.73 AP). Here, less texture information and less training data in comparison to the class car are attributed to these

results. If we applied a high confidence threshold of 0.80 in the RGB detector, we increased the precision from 0.72 to 0.84, but the mAP dropped to 0.44. The fusion with IFCNN [53] achieved a mAP of 0.45, for the class car an AP of 0.84, and for the bus 0.99. In total, we were worse with IFCNN [53] compared to our RGB baseline: The visually smaller classes pedestrian (0.22 AP), bicycle (0.04 AP), and motorcycle (0.05 AP) couldn’t be recognized by IFCNN [53]. The early fusion methods EF-DSEC and EF-1 didn’t produce satisfactory results. They could only detect large visual buses with an AP of 0.67 and 0.61. The other classes weren’t detectable. We suspect that the domain gap in the camera perspective leads to insufficient results for EF-DSEC. However, the achieved mAP of 0.50 from EF-2 is significantly better than EF-1 and EF-DSEC, and object detection of cars (0.89 AP) and buses (0.94 AP) was reliable. Furthermore, in Figure 8, we noticed that EF-1 cannot detect objects that are not in the view of both sensors. We conclude that considering objects in the training process that are not recognizable by a specific sensor domain is important. In the next step, we want to discuss our late fusion methods: SLF significantly outperformed all other fusion methods with its 0.78 mAP and the RGB baseline with more than +0.09 mAP. The recognition rate is high in all object classes. This result shows the strengths of the event-based and RGB cameras when combined. Furthermore, our late fusion STLF achieved an mAP of 0.59. Therefore, it outperforms the early fusion methods. The described results of all listed methods can tend to be observed in all traffic scenarios of the scenario “Day”.

Scenario “N-1” was recorded at night with street lights on, including the car and bus object classes. Despite a noticeable drop in recall, we can still detect cars (0.71 AP) and buses (0.80 AP) with the RGB detector. Interestingly, as shown in Figure 8, false positive detections occur significantly more often with the RGB detector. Using a higher confidence threshold could solve this problem but also result in less recall. A fusion between an event-based and RGB camera could help. As in the scenario “Day,” EF-DSEC and EF-1 didn’t allow proper object detection. Although EF-2 performed significantly better than EF-1, we achieved just 0.38 AP for the class car and 0.27 AP for the class bus and, therefore, had significant losses compared to the “Day.” In addition, EF-2 didn’t achieve the performance of IFCNN [53], which can detect cars with 0.47 AP and buses with 0.39 AP. Compared to the RGB detector, our SLF method achieved equivalent detection performance for the class car with an AP of 0.70 and an adequate AP of 0.62 for the class bus. Although the STLF method reached a lower mAP of 0.42, we could effectively eliminate false positives, see Figure 8.

The scenario “N-2” is the most challenging for object detection. We recorded this scenario in complete darkness with no street lights at around 2:00 am. Due to the low traffic volume, this subset only contains passing cars. Compared to the scenario “Day,” the performance of the RGB detector dropped for cars from 0.94 AP to 0.43 AP. The IFCNN fusion [53], EF-DSEC, EF-1, and EF-2 didn’t allow the detection of passing cars, too. According to Table V and Figure 8, the event-based detector EB could still deliver adequate or better

TABLE VI: We evaluated our detectors and fusion methods on the TUMTraF Event test set with the ‘‘L-RGB’’ labels. We distinguish between ‘‘Day,’’ night with street lights on (‘‘N-1’’), and night with street lights off (‘‘N-2’’). Furthermore, if available, we analyzed the dominant traffic flows, ‘‘Standing,’’ ‘‘Vertical,’’ and ‘‘Horizontal.’’ We used average precision, precision, and recall (AP, P, R) as metrics for the RGB detector, the IFCNN fusion [53], and the early fusion detectors EF-DSEC, which is trained on the DSEC-Detection Dataset [19]–[21], EF-1, and EF-2. To evaluate our Simple Late Fusion (SLF) and Spatiotemporal Late Fusion (STLF), we used the average precision (AP). Thresholds: Confidence = 0.3; IoU = 0.45; STLF Confidence = 0.77. The performance of the RGB detector, with a confidence threshold of 0.80, was also examined (RGB-0.80). The poor performance of EF-DSEC and EF-1 are noteworthy. The drop in performance with STLF is due to the more strict fusion logic. However, SLF significantly outperforms the RGB detector in the subsets ‘‘Day’’ and ‘‘N-2.’’ The best AP values are highlighted.

Testset		Class	RGB	RGB-0.80	IFCNN	EF-DSEC	EF-1	EF-2	SLF	STLF	
Day	Full	Pedestrian	0.64 0.91 0.66	0.17 0.97 0.17	0.22 0.94 0.22	0.03 0.53 0.03	0.00 0.00 0.00	0.24 0.94 0.25	<b>0.71</b>	0.26	
		Bicycle	0.73 0.71 0.77	0.38 0.98 0.38	0.04 1.00 0.04	0.00 0.00 0.00	0.00 0.00 0.00	0.25 0.90 0.25	<b>0.74</b>	0.55	
		Car	0.94 0.95 0.95	0.83 0.99 0.83	0.84 0.96 0.84	0.21 0.85 0.21	0.10 0.37 0.15	0.89 0.96 0.90	<b>0.95</b>	0.86	
		Motorcycle	0.37 0.56 0.48	0.20 0.44 0.19	0.05 1.00 0.05	0.00 0.00 0.00	0.00 0.00 0.00	0.10 0.67 0.10	<b>0.75</b>	0.57	
		Bus	0.98 0.74 1.00	0.92 0.91 0.91	<b>0.99</b> 0.65 1.00	0.67 0.78 0.67	0.61 0.82 0.61	0.94 0.69 0.98	<u>0.98</u>	0.97	
		Truck	0.58 0.48 0.76	0.36 0.76 0.39	0.42 0.35 0.52	0.01 0.25 0.02	0.05 0.25 0.08	0.50 0.47 0.61	<b>0.63</b>	0.47	
		Trailer	0.60 0.69 0.74	0.24 0.80 0.28	0.58 0.82 0.69	0.00 0.00 0.00	0.26 0.65 0.33	0.55 0.80 0.67	<b>0.71</b>	0.46	
		∅ Total	0.69 0.72 0.77	0.44 0.84 0.45	0.45 0.82 0.48	0.13 0.34 0.13	0.15 0.30 0.17	0.50 0.78 0.54	<b>0.78</b>	0.59	
	Standing	Pedestrian	0.70 0.95 0.68	0.18 0.97 0.18	0.23 0.96 0.23	0.04 0.56 0.05	0.00 0.01 0.00	0.27 0.98 0.27	<b>0.77</b>	0.29	
		Bicycle	<b>0.87</b> 0.72 0.87	0.52 0.98 0.53	0.06 1.00 0.05	0.00 0.00 0.00	0.00 0.00 0.00	0.34 0.93 0.35	<b>0.87</b>	0.71	
		Car	0.93 0.96 0.93	0.83 0.98 0.83	0.80 0.96 0.81	0.26 0.82 0.27	0.07 0.35 0.11	0.89 0.96 0.89	<b>0.94</b>	0.86	
		Bus	0.99 0.96 1.00	0.89 1.00 0.89	0.99 0.93 1.00	0.65 0.70 0.65	0.62 0.80 0.62	0.99 0.90 1.00	<b>1.00</b>	<b>1.00</b>	
		Truck	0.72 0.68 0.79	0.40 0.92 0.40	0.58 0.42 0.64	0.01 0.24 0.03	0.00 0.10 0.03	0.68 0.63 0.71	<b>0.78</b>	0.52	
		Trailer	0.56 0.70 0.70	0.13 0.61 0.20	0.61 0.76 0.76	0.00 0.00 0.00	0.20 0.52 0.25	0.58 0.75 0.74	<b>0.73</b>	0.32	
		∅ Total	0.80 0.83 0.83	0.49 0.91 0.50	0.55 0.84 0.58	0.16 0.39 0.17	0.15 0.30 0.17	0.63 0.86 0.66	<b>0.85</b>	0.62	
		Vertical	Pedestrian	0.44 0.78 0.47	0.13 0.92 0.12	0.21 0.86 0.23	0.00 0.00 0.00	0.00 0.00 0.00	0.14 0.72 0.19	<b>0.53</b>	0.19
	Bicycle		<b>0.48</b> 1.00 0.47	0.06 1.00 0.06	0.00 0.00 0.00	0.00 0.00 0.00	0.00 0.00 0.00	0.03 1.00 0.03	<u>0.46</u>	0.21	
	Car		<b>0.96</b> 0.95 0.96	0.85 0.99 0.86	0.87 0.95 0.88	0.18 0.85 0.19	0.13 0.35 0.18	0.90 0.95 0.91	<b>0.96</b>	<u>0.90</u>	
	Motorcycle		0.37 0.62 0.47	0.20 0.44 0.19	0.05 1.00 0.05	0.00 0.00 0.00	0.00 0.00 0.00	0.10 1.00 0.10	<b>0.75</b>	<u>0.57</u>	
	Truck		0.42 0.37 0.68	0.30 0.57 0.37	0.12 0.23 0.29	0.00 0.00 0.00	0.13 0.44 0.18	0.19 0.28 0.42	<b>0.46</b>	0.39	
	Trailer		0.67 0.75 0.70	0.38 1.00 0.38	0.60 0.93 0.61	0.00 0.00 0.00	0.36 0.80 0.42	0.56 0.92 0.58	<b>0.72</b>	0.63	
	∅ Total		0.56 0.75 0.62	0.32 0.82 0.33	0.31 0.66 0.34	0.03 0.14 0.03	0.10 0.26 0.13	0.32 0.81 0.37	<b>0.65</b>	0.48	
	Horiz.		Pedestrian	<b>0.13</b> 1.00 0.13	0.00 0.00 0.00	0.00 0.00 0.00	0.00 0.00 0.00	0.00 0.00 0.00	0.03 1.00 0.03	<b>0.13</b>	0.00
		Car	<b>0.95</b> 0.98 0.95	0.78 1.00 0.78	0.85 1.00 0.85	0.19 0.93 0.19	0.13 0.58 0.16	0.86 0.99 0.86	<b>0.95</b>	0.79	
Bus		<b>1.00</b> 1.00 1.00	0.96 1.00 0.95	<b>1.00</b> 1.00 1.00	0.71 1.00 0.70	0.60 0.86 0.60	0.96 1.00 0.95	<u>0.99</u>	0.95		
∅ Total		<b>0.69</b> 0.99 0.69	0.58 0.67 0.58	<u>0.62</u> 0.67 0.62	0.30 0.64 0.30	0.24 0.48 0.25	<u>0.62</u> 1.00 0.61	<b>0.69</b>	0.58		
N-1	Full	Car	<b>0.71</b> 0.91 0.56	0.22 0.98 0.22	0.47 0.79 0.53	0.02 0.75 0.02	0.03 0.31 0.08	0.38 0.74 0.46	<u>0.70</u>	0.41	
		Bus	<b>0.80</b> 0.62 0.83	0.57 0.96 0.57	0.39 0.45 0.55	0.00 0.00 0.00	0.10 1.00 0.10	0.27 0.58 0.33	<u>0.62</u>	0.43	
		∅ Total	<b>0.76</b> 0.77 0.70	0.40 0.97 0.40	0.43 0.62 0.54	0.01 0.38 0.01	0.07 0.66 0.09	0.33 0.66 0.40	<u>0.66</u>	0.42	
	Ver.	Car	<b>0.72</b> 0.88 0.72	0.20 0.99 0.20	0.48 0.83 0.52	0.02 0.76 0.02	0.02 0.27 0.07	0.43 0.80 0.49	<u>0.71</u>	0.38	
		Bus	0.05 0.07 0.50	0.00 0.00 0.00	0.04 0.06 0.18	0.00 0.00 0.00	0.00 0.00 0.00	<u>0.17</u> 0.10 0.17	<u>0.13</u>	<b>0.28</b>	
		∅ Total	0.39 0.48 0.61	0.10 0.50 0.10	0.26 0.45 0.35	0.01 0.38 0.01	0.01 0.14 0.04	0.30 0.45 0.33	<b>0.42</b>	0.33	
	Hor.	Car	<b>0.70</b> 0.93 0.54	0.33 0.94 0.34	0.51 0.66 0.60	0.03 1.00 0.02	0.09 0.59 0.14	0.19 0.43 0.31	<u>0.66</u>	0.58	
		Bus	<b>0.91</b> 0.88 0.86	0.67 0.96 0.67	0.48 0.64 0.60	0.00 0.00 0.00	0.12 1.00 0.11	0.35 0.93 0.36	<u>0.71</u>	0.45	
		∅ Total	<b>0.81</b> 0.91 0.70	0.50 0.95 0.50	0.49 0.65 0.60	0.01 0.50 0.01	0.11 0.80 0.12	0.27 0.68 0.33	<u>0.69</u>	0.52	
	N-2	Full	Car	0.43 0.85 0.38	0.13 1.00 0.12	0.04 0.46 0.07	0.00 0.00 0.00	0.01 0.25 0.04	0.07 0.38 0.13	<b>0.49</b>	0.27
			∅ Total	0.43 0.85 0.38	0.13 1.00 0.12	0.04 0.46 0.07	0.00 0.00 0.00	0.01 0.25 0.04	0.07 0.38 0.13	<b>0.49</b>	0.27
			Car	0.42 0.83 0.38	0.14 1.00 0.13	0.03 0.43 0.04	0.00 0.00 0.00	0.02 0.25 0.05	0.07 0.36 0.13	<b>0.49</b>	0.26
Ver.		∅ Total	0.42 0.83 0.38	0.14 1.00 0.13	0.03 0.43 0.04	0.00 0.00 0.00	0.02 0.25 0.05	0.07 0.36 0.13	<b>0.49</b>	0.26	
		Car	0.48 0.87 0.50	0.08 1.00 0.07	0.13 0.60 0.21	0.00 0.00 0.00	0.00 0.00 0.00	0.11 0.50 0.14	<b>0.61</b>	0.35	
		∅ Total	0.48 0.87 0.50	0.08 1.00 0.07	0.13 0.60 0.21	0.00 0.00 0.00	0.00 0.00 0.00	0.11 0.50 0.14	<b>0.61</b>	0.35	

detection results than the RGB detector. Since our late fusion methods, SLF and STLF, in principle, represent a logical OR of both sensors, we could achieve significant improvements in this challenging illumination scenario: The SLF achieved 0.49 mAP during the dark night and, therefore, slightly better than the RGB detector. Furthermore, we recognize higher detection performance in horizontal traffic scenarios. The SLF outperformed in this case with more than +0.13 mAP the RGB detector and achieved an mAP of 0.61. However, STLF reaches there a lower mAP value of 0.35.

Interestingly, in almost all scenarios, the performance of STLF is slightly weaker compared to the SLF method. This effect occurs due to the fact that STLF accepts only objects that either have at least a confidence threshold in the RGB

camera detection, e.g., of  $C = 0.77$ , or have been recognized over time by the event-based camera. In challenging illumination conditions, e.g., at night, where the RGB confidence value is lower, predominantly moving objects are more considered for fusion. Therefore, STLF is particularly suitable for highly dynamic traffic situations, e.g., on a motorway. Figure 9 shows this effect: Cars are slightly less detected in STLF, but higher precision values are guaranteed. This result is particularly noticeable in scenarios with challenging illumination, where the RGB camera can no longer deliver high confidence values. For example, SLF’s minimum possible precision value in scenario ‘‘N-1’’ is 0.76, whereas STLF’s is 0.90. With it, the algorithm prevents false positives, see Figure 8.

The question arises regarding which confidence threshold  $C$

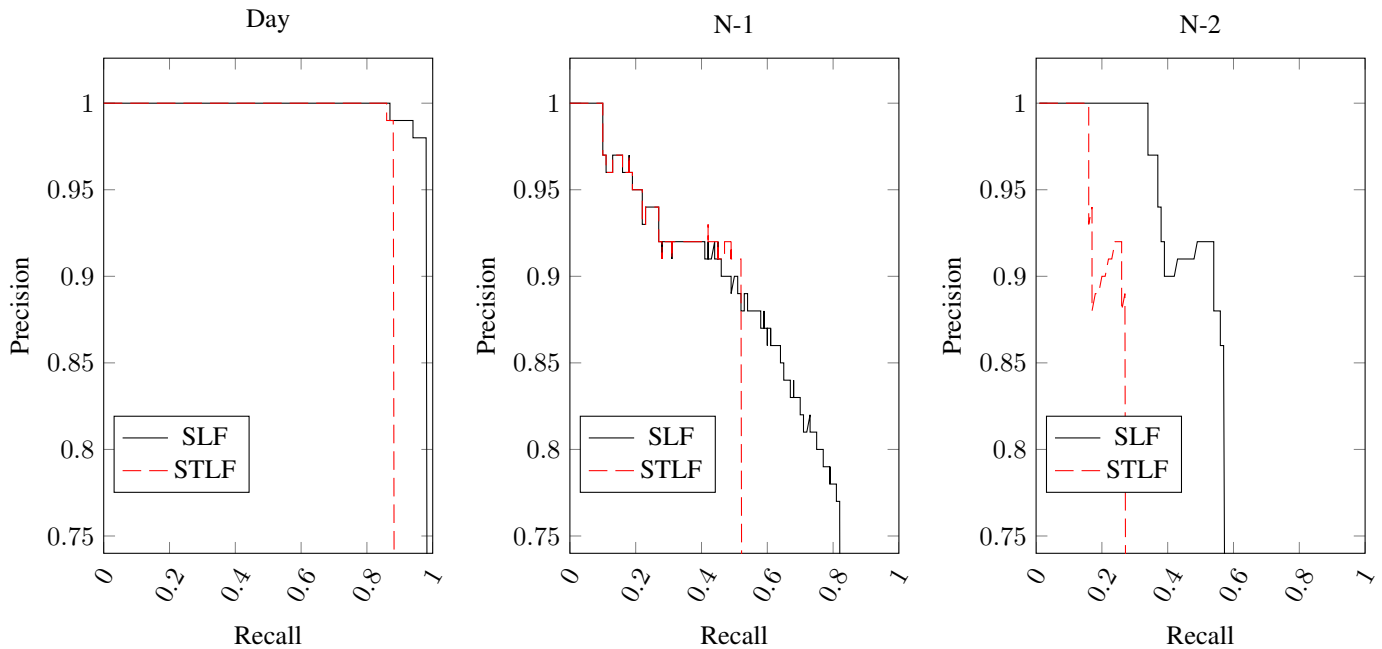


Fig. 9: The effect of Spatiotemporal Late Fusion (STLF) becomes clear on the precision-recall curve for the class “car:” Because STLF gives preferential treatment to higher confidence values or objects detected by the event-based camera, higher precision values are guaranteed. Since not every non-moving object is recognized in difficult visibility conditions, e.g., at night, the recall drops in such situations. On the other hand, Simple Late Fusion (SLF) enables higher recall values even in difficult visibility conditions with the compromise of lower precision values. We chose the class “car” because it is available in all illumination scenarios.

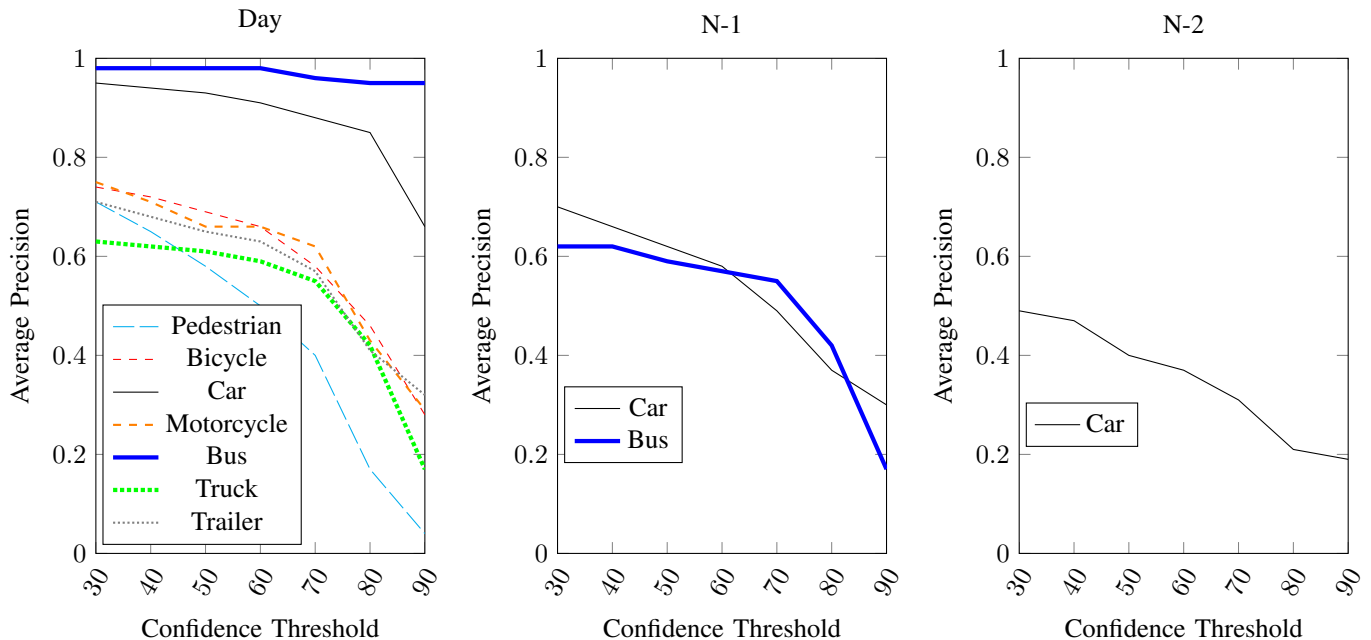


Fig. 10: This figure shows the average precision (AP) of the Spatiotemporal Late Fusion (STLF) for several confidence thresholds  $C$  in the illumination scenarios “Day,” “N-1,” and “N-2.” The detection performance on the “Day” for most classes decreases at  $C = 70$ . However, there is no performance drop for buses; the performance drop for cars occurs at  $C = 80$ . In the scenarios “N-1,” the performance drop starts at  $C = 70$ . A linear decline of the performance is recognizable in “N-2.”

offers the best detection performance with maximum filtering. For investigation, we analyzed the average precision (AP) of each available object class with different confidence values in the illumination scenarios “Day,” “N-1,” and “N-2,” see Figure 10. We observed that on the “Day,” the performance drop for cars occurred at  $C = 80$ . However, there is no drop for the class bus. For most other classes, the collapse started at  $C = 70$ . These results are plausible because, due to their geometric size, cars and buses are much easier to recognize in the RGB camera than, for example, pedestrians or cyclists. In addition, motorized vehicles generate significantly more movement in the event-based image, which leads to more trustworthy classifications. In scenario “N-1,” we also recognized a drop in detection performance starting with a confidence threshold of  $C = 70$ . In absolute darkness, as in scenario “N-2,” the detections have lower confidence values, and the detection performance decreases almost linearly. According to Figure 8, with confidence  $C = 77$ , we achieved a sufficient compromise between detection performance and reduction of false positives.

In summary, the sensor fusion between event-based and RGB cameras, particularly Simple Late Fusion (SLF), offers comparable or significantly more stable detection performance than a single sensor. Here, the strengths of both sensor types are combined: The RGB camera provides color and detailed texture information, whereas the event-based camera is primarily characterized by a high dynamic range and an extremely fast response time, which reduces motion blur. Compared to a single RGB camera, we could improve the detection performance by +0.09 mAP during the day and +0.13 mAP during the dark night. Using Spatiotemporal Late Fusion (STLF), we were also able to noticeably reduce the number of false positives. Thus, the sensor fusion with the event-based camera supplements the RGB camera and ensures more stability and reliability in object detection for a roadside ITS, even in highly challenging illumination conditions.

## V. CONCLUSION

In this work, we have improved our previous approach from [31] for targetless extrinsic calibration between event-based and RGB cameras. To find image correspondences, we have extended the matching algorithm with DBSCAN clustering [37]. This enables us to handle multiple moving objects. Furthermore, we published the novel TUMTraf Event Dataset, which contains spatiotemporal calibrated event-based and RGB images from stationary roadside cameras in the domain of Intelligent Transportation Systems. In addition, we have created an event-based image detector and the early fusion methods EF-1 and EF-2 with our dataset. Last but not least, we have developed Simple Late Fusion and Spatiotemporal Late Fusion to combine the strengths of event-based and RGB cameras with parallel reduction of their limitations.

We demonstrated a significant increase in the flexibility of our calibration approach, as it can now also be used in more complex scenarios with independent moving traffic participants. In addition, we have pointed out the advantages and disadvantages of event-based and RGB cameras in the

domain of roadside ITS in several traffic scenarios during the day and at night. Here, we showed that our presented sensor fusion algorithms can significantly increase the detection performance compared to an RGB camera of up to +9% mAP during the day and up to +13% mAP at night. Furthermore, we compared our results qualitatively and quantitatively with those from other fusion methods and event-based camera datasets. In applying a roadside Intelligent Transportation System, our methods delivered significantly more convincing results, which underlines the relevance of our methods and our TUMTraf Event Dataset.

For future work, we propose significantly expanding the TUMTraf Event Dataset. Thus, more accurate object detection could be possible during the day and at night. This goal can be achieved in a highly scalable manner using so-called pseudo-labeling with our targetless calibration tool. Furthermore, developing a trustworthy artificial intelligence, e.g., using the framework of [48], [49], would be a valuable and necessary step towards safe autonomous driving supported by Intelligent Transportation Systems. Last but not least, instead of DBSCAN [37] clusterization, we propose deep-learning-based targetless calibration, which is now possible with our TUMTraf Event Dataset.

## ACKNOWLEDGMENT

This research was funded by the Federal Ministry of Education and Research of Germany in the project AUTOtech.agil, FKZ: 01IS22088U. We would like to express our gratitude for making this paper possible.

## REFERENCES

- [1] G. Chen, H. Cao, J. Conrads, H. Tang, F. Rohrbein, and A. Knoll, “Event-based neuromorphic vision for autonomous driving: A paradigm shift for bio-inspired visual sensing and perception,” *IEEE Signal Processing Magazine*, vol. 37, no. 4, pp. 34–49, 2020.
- [2] G. Gallego, T. Delbruck, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. J. Davison, J. Conrads, K. Daniilidis, and D. Scaramuzza, “Event-based vision: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 1, pp. 154–180, 2022.
- [3] C. Creß, Z. Bing, and A. C. Knoll, “Intelligent transportation systems using roadside infrastructure: A literature survey,” *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–0, 2023.
- [4] G. Chen, H. Cao, M. Aafaque, J. Chen, C. Ye, F. Röhrebein, J. Conrads, K. Chen, Z. Bing, X. Liu, G. Hinz, W. Stechele, and A. Knoll, “Neuromorphic vision based multivehicle detection and tracking for intelligent transportation system,” *Journal of Advanced Transportation*, vol. 2018, pp. 1–13, 2018.
- [5] G. Chen, H. Cao, C. Ye, Z. Zhang, X. Liu, X. Mo, Z. Qu, J. Conrads, F. Röhrebein, and A. Knoll, “Multi-cue event information fusion for pedestrian detection with neuromorphic vision sensors,” *Frontiers in Neurobotics*, vol. 13, p. 10, 2019.
- [6] H. Cao, G. Chen, J. Xia, G. Zhuang, and A. Knoll, “Fusion-based feature attention gate component for vehicle detection based on event camera,” *IEEE Sensors Journal*, vol. 21, no. 21, pp. 24 540–24 548, 2021.
- [7] J. Zhang, Y. Wang, W. Liu, M. Li, J. Bai, B. Yin, and X. Yang, “Frame-event alignment and fusion network for high frame rate tracking,” in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 6/17/2023 - 6/24/2023, pp. 9781–9790.
- [8] Z. Zhou, Z. Wu, R. Boutteau, F. Yang, C. Démonceaux, and D. Ginjac, “Rgb-event fusion for moving object detection in autonomous driving,” in *ICRA 2023*, M. O’Malley, Ed. Piscataway, NJ: IEEE, 2023, pp. 7808–7815.
- [9] A. Tomy, A. Paigwar, K. S. Mann, A. Renzaglia, and C. Laugier, “Fusing event-based and rgb camera for robust object detection in adverse conditions,” in *2022 IEEE International Conference on Robotics and Automation (ICRA)*, M. O’Malley, Ed. IEEE: IEEE, 2022, pp. 933–939.



- [10] H. Sun and V. Fremont, "Object tracking with a fusion of event-based camera and frame-based camera," in *Intelligent Systems and Applications*, ser. Lecture Notes in Networks and Systems Ser. K. Arai, Ed. Cham: Springer International Publishing AG, 2023, vol. 543, pp. 250–264.
- [11] J. Binias, D. Neil, S.-C. Liu, and T. Delbruck, "Ddd17: End-to-end davis driving dataset." [Online]. Available: <https://arxiv.org/pdf/1711.01458.pdf>
- [12] E. Mueggler, H. Rebecq, G. Gallego, T. Delbruck, and D. Scaramuzza, "The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam," *The International Journal of Robotics Research*, vol. 36, no. 2, pp. 142–149, 2017.
- [13] A. Sironi, M. Brambilla, N. Bourdis, X. Lagorce, and R. Benosman, "Hats: Histograms of averaged time surfaces for robust event-based object classification," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE, 2018, pp. 1731–1740.
- [14] A. Z. Zhu, D. Thakur, T. Ozaflan, B. Pfommer, V. Kumar, and K. Daniilidis, "The multivehicle stereo event camera dataset: An event camera dataset for 3d perception," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2032–2039, 2018.
- [15] W. Cheng, H. Luo, W. Yang, L. Yu, S. Chen, and W. Li, "Det: A high-resolution dvs dataset for lane extraction," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition workshops*. Piscataway, NJ: IEEE, 2019, pp. 1666–1675.
- [16] Y. Hu, J. Binias, D. Neil, S.-C. Liu, and T. Delbruck, "Ddd20 end-to-end event camera driving dataset: Fusing frames and events with deep learning for improved steering prediction." [Online]. Available: <http://arxiv.org/pdf/2005.08605.pdf>
- [17] E. Perot, P. de Tournemire, D. Nitti, J. Masci, and A. Sironi, "Learning to detect objects with a 1 megapixel event camera," *NIPS'20: Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020. [Online]. Available: <http://arxiv.org/pdf/2009.13436.pdf>
- [18] P. de Tournemire, D. Nitti, E. Perot, D. Migliore, and A. Sironi, "A large scale event-based detection dataset for automotive." [Online]. Available: <http://arxiv.org/pdf/2001.08499.pdf>
- [19] M. Gehrig, W. Aarents, D. Gehrig, and D. Scaramuzza, "Dsec: A stereo event camera dataset for driving scenarios," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4947–4954, 2021.
- [20] D. Gehrig and D. Scaramuzza, "Pushing the limits of asynchronous graph-based object detection with event cameras." [Online]. Available: <http://arxiv.org/pdf/2211.12324.pdf>
- [21] "Dsec-detection - dsec," 27/02/2024. [Online]. Available: <https://dsec.ifi.uzh.ch/dsec-detection/>
- [22] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza, "Esim: an open event camera simulator," *Conference on Robot Learning*, pp. 969–982, 2018. [Online]. Available: <https://proceedings.mlr.press/v87/rebecq18a.html>
- [23] N. F. Y. Chen, "Pseudo-labels for supervised learning on dynamic vision sensor data, applied to object detection under ego-motion," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition workshops*. Piscataway, NJ: IEEE, 2018, pp. 757–75709.
- [24] M. Muglikar, M. Gehrig, D. Gehrig, and D. Scaramuzza, "How to calibrate your event camera," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2021.
- [25] K. Huang, Y. Wang, and L. Kneip, "Dynamic event camera calibration," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021.
- [26] Qifan Zhang, Jinwei Ye, Philip Osteen, and S Susan Young, *Co-Calibration and Registration of Color and Event Cameras*, 2020. [Online]. Available: <https://apps.dtic.mil/sti/citations/AD1116489>
- [27] M. J. Dominguez-Morales, A. Jimenez-Fernandez, G. Jimenez-Moreno, C. Conde, E. Cabello, and A. Linares-Barranco, "Bio-inspired stereo vision calibration for dynamic vision sensors," *IEEE Access*, vol. 7, pp. 138 415–138 425, 2019.
- [28] E. Mueggler, N. Baumli, F. Fontana, and D. Scaramuzza, "Towards evasive maneuvers with quadrotors using dynamic vision sensors," in *2015 European Conference on Mobile Robots (ECMR)*, 2015, pp. 1–8.
- [29] C. Plasberg, M. G. Besselmann, A. Roennau, and R. Dillmann, "Intrinsic and extrinsic calibration method for a trinocular multimodal camera setup," in *2022 25th International Conference on Information Fusion (FUSION)*. IEEE, 7/4/2022 - 7/7/2022, pp. 1–6.
- [30] E. Mueggler, B. Huber, and D. Scaramuzza, "Event-based, 6-dof pose tracking for high-speed maneuvers," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 2761–2768.
- [31] C. Creß, E. Schütz, B. L. Žagar, and A. C. Knoll, "Targetless extrinsic calibration between event-based and rgb camera for intelligent transportation systems," in *2023 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 6/4/2023 - 6/7/2023, pp. 1–8.
- [32] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Uppcroft, "Simple online and realtime tracking," in *2016 IEEE International Conference on Image Processing*. Piscataway, NJ: IEEE, 2016, pp. 3464–3468.
- [33] innovation mobility.com, "Providentia++: A9 testfeld für autonomes fahren und digitale erkennung von fahrzeugen," 18/08/2022. [Online]. Available: <https://innovation-mobility.com/projekt-providentia/>
- [34] arXiv.org, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 07/07/2022.
- [35] W. Kin-Yiu, "Implementation of paper - yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 14/01/2023. [Online]. Available: <https://github.com/WongKinYiu/yolov7/tree/mask>
- [36] H. Chen, K. Sun, Z. Tian, C. Shen, Y. Huang, and Y. Yan, "Blendmask: Top-down meets bottom-up for instance segmentation." [Online]. Available: <https://arxiv.org/pdf/2001.00309>
- [37] Ester, Martin and Kriegel, Hans-Peter and Sander, Jörg and Xu, Xiaowei, "A density-based algorithm for discovering clusters in large spatial databases with noise," *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pp. 226–231, 1996.
- [38] G. Simarro, D. Calvete, and P. Souto, "Ucalib: Cameras autocalibration on coastal video monitoring systems," *Remote Sensing*, vol. 13, no. 14, p. 2795, 2021.
- [39] T. Fu, H. Yu, W. Yang, Y. Hu, and S. Scherer, "Targetless extrinsic calibration of stereo cameras, thermal cameras, and laser sensors in the wild." [Online]. Available: <https://arxiv.org/pdf/2109.13414>
- [40] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004. [Online]. Available: <https://link.springer.com/article/10.1023/B:VISI.0000029664.99615.94>
- [41] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.
- [42] A. Tsaregorodtsev, J. Müller, J. Strohecker, M. Herrmann, M. Buchholz, and V. Belagiannis, "Extrinsic camera calibration with semantic segmentation." [Online]. Available: <http://arxiv.org/pdf/2208.03949v1>
- [43] M. A. Munoz-Banon, F. A. Candelas, and F. Torres, "Targetless camera-lidar calibration in unstructured environments," *IEEE Access*, vol. 8, pp. 143 692–143 705, 2020.
- [44] L. Wang, Z. Zhang, X. Di, and J. Tian, "A roadside camera-radar sensing fusion system for intelligent transportation," in *2020 17th European Radar Conference*. Piscataway, NJ: IEEE, 2021, pp. 282–285.
- [45] A. Sengupta, A. Yoshizawa, and S. Cao, "Automatic radar-camera dataset generation for sensor-fusion applications," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2875–2882, 2022.
- [46] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *29th IEEE Conference on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE, 2016, pp. 779–788.
- [47] F.-Y. Wang, Q. Miao, L. Li, Q. Ni, X. Li, J. Li, L. Fan, Y. Tian, and Q.-L. Han, "When does sora show: The beginning of tao to imaginative intelligence and scenarios engineering," *IEEE/CAA Journal of Automatica Sinica*, vol. 11, no. 4, pp. 809–815, 2024. [Online]. Available: <https://ieeexplore.ieee.org/ielx7/6570654/10475140/10475159.pdf?tp=&arnumber=10475159&isnumber=10475140&ref=>
- [48] X. Li, Y. Tian, P. Ye, H. Duan, and F.-Y. Wang, "A novel scenarios engineering methodology for foundation models in metaverse," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 4, pp. 2148–2159, 2023.
- [49] X. Li, P. Ye, J. Li, Z. Liu, L. Cao, and F.-Y. Wang, "From features engineering to scenarios engineering for trustworthy ai: I&i, c&c, and v&v," *IEEE Intelligent Systems*, vol. 37, no. 4, pp. 18–26, 2022.
- [50] M. Brenner, N. H. Reyes, T. Susnjak, and A. L. C. Barczak, "Rgb-d and thermal sensor fusion: A systematic literature review," *IEEE Access*, vol. 11, pp. 82 410–82 442, 2023.
- [51] D. F. Crouse, "On implementing 2d rectangular assignment algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 4, pp. 1679–1696, 2016.
- [52] W. Zimmer, J. Birkner, M. Brucker, H. Tung Nguyen, S. Petrovski, B. Wang, and A. C. Knoll, "Infradet3d: Multi-modal 3d object detection based on roadside infrastructure camera and lidar sensors," in *2023 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 6/4/2023 - 6/7/2023, pp. 1–8.

- [53] Y. Zhang, Y. Liu, P. Sun, H. Yan, X. Zhao, and L. Zhang, "Ifcnn: A general image fusion framework based on convolutional neural network," *Information Fusion*, vol. 54, pp. 99–118, 2020.
- [54] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *2017 IEEE International Conference on Computer Vision*, ser. IEEE Xplore Digital Library. Piscataway, NJ: IEEE, 2017, pp. 2999–3007.
- [55] H. Rebecq, R. Ranfil, V. Koltun, and D. Scaramuzza, "High speed and high dynamic range video with an event camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 6, pp. 1964–1980, 2021.
- [56] Y. Tian, X. Li, H. Zhang, C. Zhao, B. Li, X. Wang, and F.-Y. Wang, "Vistagpt: Generative parallel transformers for vehicles with intelligent systems for transport automation," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 9, pp. 4198–4207, 2023.
- [57] M. Ross, *Modellfreie, statistische Objektverfolgung in Farbbildsequenzen: Zugl.: Koblenz-Landau, Univ., Diss., 2007*. Tönning: Der Andere Verlag, 2007.
- [58] J. Kim, X. Wang, H. Wang, C. Zhu, and D. Kim, "Fast moving object detection with non-stationary background," *Multimedia Tools and Applications*, vol. 67, no. 1, pp. 311–335, 2013. [Online]. Available: <https://link.springer.com/article/10.1007/s11042-012-1075-3>
- [59] Jianbo Shi and Tomasi, "Good features to track," in *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1994, pp. 593–600.
- [60] J. Bouguet, "Pyramidal implementation of the lucas kanade feature tracker," 1999. [Online]. Available: <https://www.semanticscholar.org/paper/Pyramidal-implementation-of-the-lucas-kanade-Bouguet/aa972b40c0f8e20b07e02d1fd320bc7ebadfdc7?p2df>
- [61] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern Recognition Letters*, vol. 27, no. 7, pp. 773–780, 2006.
- [62] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context." [Online]. Available: <http://arxiv.org/pdf/1405.0312.pdf>
- [63] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.
- [64] M. A. Fischler and R. C. Bolles, "Random sample consensus," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [65] 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," *Field Robotics*, vol. 2, no. 1, pp. 1156–1176, 2022.
- [66] PROVIDENTIA – A9 Testfeld für autonomes Fahren und digitale Erkennung von Fahrzeugen, "Providentia++: Bmdv-projekt schafft digitalen zwilling," 15/02/2023. [Online]. Available: <https://innovation-mobility.com/projekt-providentia/>
- [67] Motional, "nuimages," 08/08/2023. [Online]. Available: <https://www.nuscenes.org/nuimages>
- [68] Creß, Christian and Zimmer, Walter and Strand, Leah and Fortkord, Maximilian and Dai, Siyi and Lakshminarasimhan, Venkatnarayanan and Knoll, Alois, "A9-dataset: Multi-sensor infrastructure-based dataset for mobility research," in *2022 IEEE Intelligent Vehicles Symposium (IV)*, 2022, pp. 965–970.
- [69] W. Zimmer, C. Creß, H. T. Nguyen, and A. C. Knoll, "Tumtraf intersection dataset: All you need for urban 3d camera-lidar roadside perception," in *2023 IEEE Intelligent Transportation Systems (ITSC)*. IEEE, 2023.
- [70] "Openlabel concept paper," 31/12/2023. [Online]. Available: <https://www.asam.net/index.php?eID=dumpFile&t=f&f=3876&token=413e8c85031ae64cc35cf42d0768627514868b2f>
- [71] C. Yang, P. Liu, G. Chen, Z. Liu, Y. Wu, and A. Knoll, "Event-based driver distraction detection and action recognition," in *2022 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. Piscataway, NJ: IEEE, 2022, pp. 1–7.
- [72] R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. Da Silva, "A comparative analysis of object detection metrics with a companion open-source toolkit," *Electronics*, vol. 10, no. 3, p. 279, 2021.



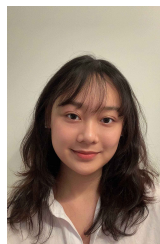
**Christian Creß** joined the Chair of Robotics, Artificial Intelligence, and Real-time Systems at the Technical University of Munich (TUM), Germany, in 2020 as a Research Assistant and Ph.D. student, where he is currently researching computer vision and data fusion for multi-modal sensor systems. He completed his M.Sc. in Applied Computer Science at the University of Applied Sciences Kempten in 2016. The Master's thesis was in computer vision and machine learning. His further research interests are artificial intelligence and software architecture.



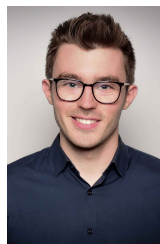
**Walter Zimmer** is currently a Ph.D. candidate and Research Assistant at the Chair of Robotics, Artificial Intelligence, and Real-time Systems of the Technical University of Munich (TUM). He received his M.Sc. degree in Computer Science from the TUM in 2018. During his studies, he stayed abroad at the Technical University of Delft (Netherlands) and the University of California, San Diego (UCSD), where he developed perception and autonomous driving algorithms. His research interests are 3D perception, simulation, and autonomous driving.



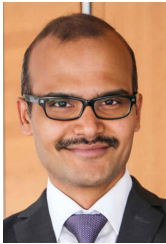
**Nils Purschke** is a Ph.D. student at the Technical University of Munich in Germany. He started his Ph.D. in 2023 at the Chair of Robotics, Artificial Intelligence, and Real-time Systems and conducted research in autonomous driving, vehicle software architecture, and artificial intelligence. He completed his Bachelor's degree (2021) and Master's degree (2023) at the Technical University of Darmstadt. In his Master's thesis, he created 3D digital twins. Parallel to his scientific work, he has been working as a software developer in IT security since 2020.



**Bach Ngoc Doan** began her B.Sc. in Informatics at the Technical University of Munich (TUM), Germany, in 2020. She is currently working on her Bachelor's thesis at the Chair of Robotics, Artificial Intelligence, and Real-time Systems at TUM. Her thesis is in the field of computer vision and focuses on monocular 3D perception for autonomous driving.



**Sven Kirchner** joined the Chair of Robotics, Artificial Intelligence, and Real-time Systems at the Technical University of Munich (TUM) in Germany as a Ph.D. Student in 2023. He received his B.Sc. and M.Sc. in Automotive Engineering from RWTH Aachen University. During his studies, he focused on the control of automated vehicles and automotive systems engineering. His research interests include autonomous driving, safe architectures for software-defined vehicles, and artificial intelligence.



**Venkatnarayanan Lakshminarasimhan** is a Ph.D. candidate and Research Assistant at the Chair of Robotics, Artificial Intelligence, and Real-time Systems, TUM School of Computation, Information and Technology, Technical University of Munich. His current research interests include V2X communication in intelligent transportation systems.



**Leah Strand** joined the Chair of Robotics, Artificial Intelligence, and Real-time Systems at the TUM as a research assistant and a Ph.D. candidate in March 2020. She received her Master's degree in Mechatronics and Information Technology from the TUM after finishing her Bachelor's degree in Engineering Science at the TUM. Her research interests are autonomous driving, multi-object tracking, sensor calibration, and fusion.



**Alois Knoll** (Senior Member) received the diploma (M.Sc.) degree in electrical/communications engineering from the University of Stuttgart, Stuttgart, Germany, in 1985, and the Ph.D. (summa cum laude) degree in computer science from the Technical University of Berlin (TU Berlin), Berlin, Germany, in 1988. He served on the faculty of the Computer Science Department at TU Berlin until 1993. He joined the University of Bielefeld, Germany, as a Full Professor and served as the Director of the Technical Informatics Research Group until 2001.

Since 2001, he has been a Professor at the Department of Informatics, Technical University of Munich (TUM), Munich, Germany. He was also on the Board of Directors of the Central Institute of Medical Technology, TUM (IMETUM). From 2004 to 2006, he was the Executive Director of the Institute of Computer Science, TUM. Between 2007 and 2009, he was a member of the EU's highest advisory board on information technology, the Information Society Technology Advisory Group, and a member of its subgroup on Future and Emerging Technologies (FET). In this capacity, he was actively involved in developing the concept of the EU's FET Flagship projects. His research interests include cognitive, medical, and sensor-based robotics, multi-agent systems, data fusion, adaptive systems, multimedia information retrieval, model-driven development of embedded systems with applications to automotive software and electric transportation, and simulation systems for robotics and traffic.