Making Bertha See

Uwe Franke, David Pfeiffer, Clemens Rabe, Carsten Knoeppel,
Markus Enzweiler, Fridtjof Stein, and Ralf G. Herrtwich

Daimler AG - Research & Development, 71059 Sindelfingen, Germany
firstname.lastname@daimler.com

Abstract

With the market introduction of the 2014 Mercedes-Benz S-Class vehicle equipped with a stereo camera system, autonomous driving has become a reality, at least in low speed highway scenarios. This raises hope for a fast evolution of autonomous driving that also extends to rural and urban traffic situations. In August 2013, an S-Class vehicle with close-to-production sensors drove completely autonomously for about 100 km from Mannheim to Pforzheim, Germany, following the well-known historic Bertha Benz Memorial Route. Next-generation stereo vision was the main sensing component and as such formed the basis for the indispensable comprehensive understanding of complex traffic situations, which are typical for narrow European villages. This successful experiment has proved both the maturity and the significance of machine vision for autonomous driving. This paper presents details of the employed vision algorithms for object recognition and tracking, free-space analysis, traffic light recognition, lane recognition, as well as self-localization.

1. Introduction

In August 1888, Bertha Benz used the three wheeled vehicle of her husband, engineer Carl Benz, to drive from Mannheim to Pforzheim, Germany. This historic event is nowadays looked upon as the birth date of the modern automobile. Exactly 125 years later, a brand-new 2014 Mercedes-Benz S-Class named ”Bertha” repeated this journey, but this time in a fully autonomous manner, see Figure 1. Following the official *Bertha Benz Memorial Route*, this car drove through the very heart of the famous city of Heidelberg, passed the Bruchsal Castle, and crossed narrow villages in the Black Forest. It stopped in front of red traffic lights, made its way through a lot of roundabouts, planned its path through narrow passages with oncoming vehicles and numerous cars parked on the road, and gave the right of way to crossing pedestrians. While Bertha Benz wanted to demonstrate the maturity of the gasoline engine developed by her husband, the goal of our experiment was to show that autonomous driving is not limited to highways and similar well-structured environments anymore. An additional aim was to learn about situations that still cause problems, in order to identify further research directions.

Figure 1: Autonomous vehicle ”Bertha”, a 2014 Mercedes-Benz S-Class with well-integrated close-to-production sensors driving fully autonomously on open public roads.

We only carefully modified the serial-production sensor setup already available in our vehicle, as follows. Four 120° mid-range radars were added for better intersection monitoring. The baseline of the car’s existing stereo camera system was enlarged to 35 cm for increased precision and distance coverage. For traffic light recognition, self-localization and pedestrian recognition in turning maneuvers, two wide angle monocular color cameras were added.

2. Related Work

The general vision of autonomous driving has quite a long history which is well documented on the web [25]. It

Acknowledgements: The Bertha Project was a group effort. We are indebted to all project members, most notably Eberhard Kaus for his superb project management and Thao Dang for his outstanding technical guidance. Special thanks goes to Markus Schreiber for generating digital maps, Friedrich Erbs for the optimal segmentation of the dynamic Stixel-World, Frank Lindner for traffic light recognition, and Nico Schneider for testing.

first appeared in the 1970s with the idea of inductive cabling for lateral guidance. In the 1980s, the CMU vehicle Navlab drove slowly on the Pittsburgh campus using cameras. Following the introduction of Kalman Filtering for image sequence analysis, Dickmanns demonstrated vision-based lane keeping on a German highway with speeds of up to 100 km/h [3]. This seminal work represents the foundation of nearly all commercially available lane keeping systems on the market today. At the final presentation of the European PROMETHEUS project in Paris in 1994, vision-based autonomous driving on a public highway was demonstrated including lane change maneuvers [2]. In July 1995, Pommerleau (CMU) drove with the Navlab5 vehicle from Washington DC to San Diego using vision-based lateral guidance and radar-based adaptive cruise control (ACC) at an autonomy rate of 98.2 % [13, 17]. In the same year, Dickmanns' team drove approximately 1750 km from Munich, Germany, to Odense, Denmark, and back at a maximum speed of 175 km/h. The longest distance travelled without manual intervention by the driver was 158 km. On average, manual intervention was necessary every 9 km [2]. All those approaches were focused on well-structured highway scenarios, where the autonomous driving task is much easier than in constantly changing and chaotic urban traffic.

Sparked by the increased methodical and technical availability of better algorithms and sensors, initial steps towards taking autonomous driving into urban scenarios were made by Franke [10] in 1998. One notable and highly important event was the Urban Challenge in 2007. Here, all finalists based their work on high-end laser scanners coupled with radars for long range sensing. The impressive work by Google in the field of autonomous driving is based on the experience gained in the Urban Challenge. As a result, they also adopted high-end laser scanners and long-range radars as main sensing platforms in their system, augmented by a high-resolution color camera for traffic light recognition.

Very recently, on July 12th 2013, Broggi and his group performed an impressive autonomous driving experiment in Parma, Italy [24]. Their vehicle moved autonomously in public traffic, even at times with nobody in the driver's seat. The 13 km long route included two-way rural roads, two freeways with junctions, and urban areas with pedestrian crossings, tunnels, artificial bumps, tight roundabouts, and traffic lights.

3. System Design and Layout

In our vision of future autonomous driving, detailed maps will be one foundational component of the system, besides the actual active sensing modules. Given the commercial nonavailability of such high-quality maps today, they were generated by our research partner, the Karlsruhe Institute of Technology (KIT), in a semi-automatic fashion. Infrastructural elements that are relevant to our application, *e.g.* speed limits, pedestrian crossings, or stop lines, have also been included into the digital map. Similar to successful Urban Challenge approaches, optimal driving paths have been calculated in an off-line step. Given a precise ego-localization relative to the map in on-line mode, our vehicle follows the pre-planned path as long as the traffic situation permits. This planning and decision module continuously analyzes the scene content delivered by the environment perception and dynamically reacts by re-planning whenever driving paths are currently blocked or will be obstructed by other traffic participants in the future.

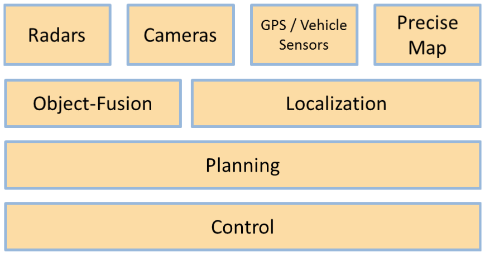


Figure 2: Software system architecture overview.

Preliminary tests proved that GPS accuracy is insufficient in most cities and villages to achieve the self-localization precision we require for the previously mentioned dynamic planning step. Hence, we combined GPS with inertial vehicle sensors for localization and additionally utilized our vision system to significantly increase self-localization precision.

Besides machine vision, which will be the main focus of the remainder of this paper, radar sensors are employed to detect moving objects at long distances as well as the surveillance of the area around the car. They ensure safe behavior at roundabouts, monitor crossing streets at intersections and are used for safe lane change maneuvers.

Figure 2 shows the layer-based software architecture of Bertha. On the sensing layer we use radars, cameras, a GPS unit coupled with inertial vehicle sensors, and a precise digital map. Object-level fusion builds a comprehensive understanding of the current traffic situation and their participants. In parallel, the visual localization results are combined with the GPS and inertial results to obtain an optimum self-localization estimate. Based on this, the planning module determines the appropriate next behavior which is then actuated by the control layer. All system modules, including our complex vision algorithms, see Section 4, operate in real-time, *i.e.* at a frame-rate of 25 Hz.

4. The Vision System

Bertha's vision system consists of three different camera setups, see Figure 3, *i.e.* one wide-angle monocular camera to recognize traffic lights and pedestrians in turning maneu-

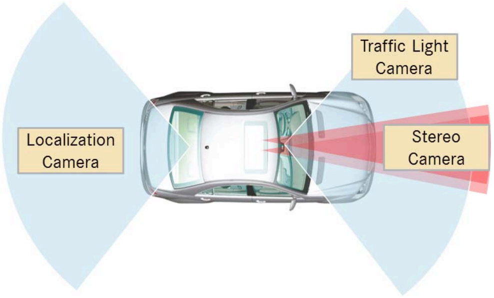


Figure 3: Overview of Bertha's camera setup. We use a stereo camera system with a 45° field-of-view (red) and two wide-angle 90° field-of-view monocular cameras (blue).

vers, another wide-angle camera for feature-based localization and a powerful stereo system for lane recognition and 3D scene analysis. The latter can be further sub-divided into three main tasks:

1. Free-space analysis: *Can Bertha drive safely along the planned path?*
2. Obstacle detection: *Are there obstacles in Bertha's path? Are they stationary or moving? What size do they have? How do they move?*
3. Object classification: *What is the type of obstacles and other traffic participants,* e.g. *pedestrians, bicyclists, or vehicles?*

Although various vision systems are already on board for advanced driver assistance, including fully autonomous emergency braking for pedestrians, the existing algorithms had to be improved significantly. The reason is that in safety critical assistance systems the vision algorithms are designed for a minimum false positive rate while keeping the true positive rate sufficiently high. An autonomous system however requires the environment perception module to detect nearly all obstacles and - at the same time - to have an extremely low false positive rate.

4.1. Lane Recognition and Localization

While lane keeping on highways and well-structured rural roads is widely available in production cars, lane recognition in cities remains an unsolved problem for several reasons: no strict rules apply for urban roads, low speeds allow for rapid changes in the lane course, lanes are marked sparsely or even not at all. However, if the markings and curbs are known from a digital map, the task of lane recognition can be reduced to a graph matching problem.

Figure 4 illustrates this principle and shows the re-projection of lane markings (blue) and curbs (yellow) from the map into the image of our stereo system. Self-localization with respect to the map involves finding the best alignment between map data and the image in terms of position and orientation. This principle has already been successfully employed in [9]. A higher robustness is achieved by using a Kalman Filter for continuous estimation of the vehicle's pose, supported by the available inertial measurements. This particularly helps with the estimation of the longitudinal position, given that this is much less defined by the markings than the lateral position. However, we found that even the longitudinal position can be reliably estimated with sufficiently precise digital maps.

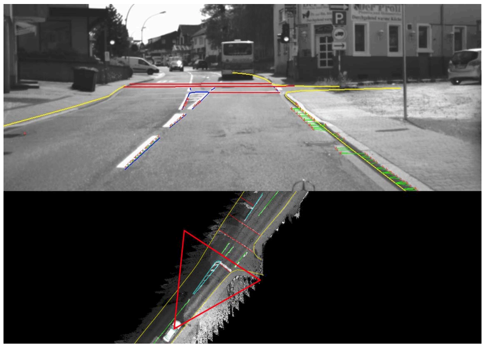


Figure 4: Re-projection of map information (bottom) into the image (top) domain.

A crucial part is the map generation process. An automatic generation using aerial images as proposed in [20] is attractive but not always possible. We decided to build our maps from a recorded video sequence using a car with a high-precision GPS localization similar to [22]. Standard postprocessing was applied to the GPS data offline in order to correct faulty traces. At this time, we manually selected markings and curbs. Obviously, this needs to be automated in the future.

In urban areas, lanes are often not marked but bounded by curbs instead. Hence, a reliable recognition of curbs is required to be able to use the map matching module. For robust curb recognition, we adopt the classification approach described in [7]. Given the expected lateral position of a curb, an appropriate image region is cropped and rectified in a way that the slanted curb becomes (nearly) vertical and independent of the distance. Then, this ROI is fed into

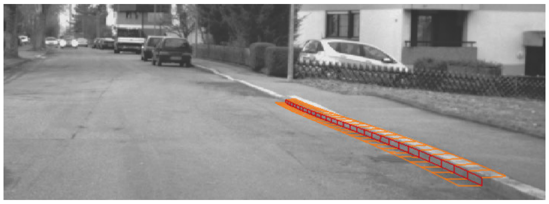
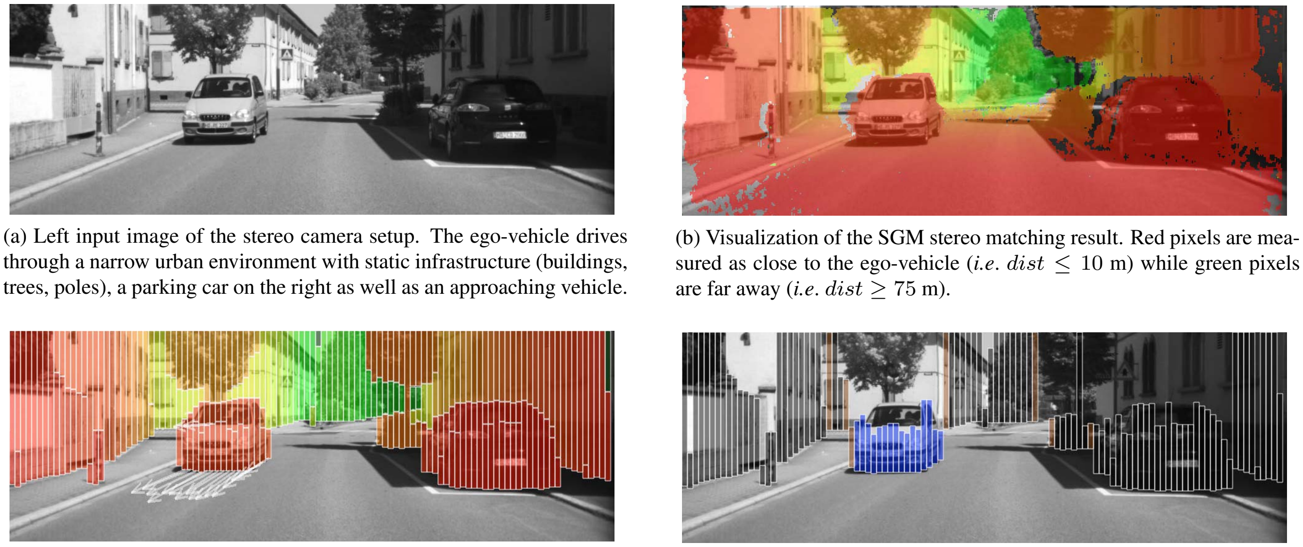


Figure 5: Example of classification-based curb recognition.



(a) Left input image of the stereo camera setup. The ego-vehicle drives through a narrow urban environment with static infrastructure (buildings, trees, poles), a parking car on the right as well as an approaching vehicle.

(b) Visualization of the SGM stereo matching result. Red pixels are measured as close to the ego-vehicle (*i.e.* $dist \leq 10$ m) while green pixels are far away (*i.e.* $dist \geq 75$ m).

(c) Stixel World representation of the disparity input. Objects are efficiently described using vertical rectangles. The arrows on the base-points of the Stixels show the estimated object velocity. The color encodes the distance.

(d) Segmentation of the Stixel World into static background/infrastructure and moving objects. The color represents a group of connected Stixels with similar motion. Brown Stixels are flagged as potentially inaccurate.

Figure 6: Visual outline of the stereo processing pipeline. Dense disparity images are computed from sequences of stereo image pairs. From this data, the Stixel World is computed, a very compact and efficient intermediate representation of the three-dimensional environment. Stixels are tracked over time for estimating the motion of other objects. This information is used to extract both static infrastructure and moving objects for subsequent processing tasks.

a multi-cue classifier operating on gray-value information and height profiles obtained from stereo vision. Through the optimal combination of both modalities, curbs are reliably detected, see Figure 5 for an example.

To further reinforce vision-based self-localization we adopt the feature-based localization approach of [15], that operates on the monocular localization camera, see Figure 3. Outside of the vision system, this additional location estimate is then fused with the map-based localization described above.

4.2. Stereo Vision

A stereo camera is used to perceive and understand the environment in front of the ego-vehicle, covering a range of up to 75 m using 1024×440 px imagers with $45°$ degree FOV lenses and a baseline of 35 cm. The stereo processing pipeline consists of four main steps: the dense stereo reconstruction itself, the Stixel segmentation, a motion estimation of other objects, and the final object segmentation. The different processing steps are briefly illustrated in Figure 6.

Stereo Matching Given the stereo image pairs, dense disparity images are reconstructed using semi-global matching (SGM) [12], *c.f.* Figure 6a and Figure 6b. This scheme was made available on an efficient, low-power FPGA-platform by [11]. The input images are processed at 25 Hz with about $400,000$ individual depth measurements per frame.

Stixel Computation To cope with this large amount of data, we utilize the Stixel representation introduced in [19]. The idea is to approximate all objects within the three-dimensional environment using sets of thin, vertically oriented rectangles, the so-called Stixels. All areas of the image that are not covered with Stixels are implicitly understood as free, and thus, in intersection with the map of the route, as potentially driveable space. To consider non-planar ground surfaces, the vertical road slope is estimated as well. Altogether, the content of the scene is represented by an average of about 300 Stixels. Just like SGM, the Stixel computation is performed on an FPGA platform.

Motion Estimation Autonomously navigating through urban environments asks for detecting and tracking other moving traffic participants, like cars or bicyclists. In our setup, this is achieved by tracking Stixels over time using Kalman filtering following the approach of [18]. Assuming a constant velocity, the motion of other objects across the ground surface is estimated for every Stixel individually. The result of this procedure is given in Figure 6c showing both the Stixel representation and the motion prediction of the Stixels.

Object Segmentation Up to this point, Stixels are processed independently, both during image segmentation and tracking. Yet, given the working principle of this representation, it is quite likely for adjacent Stixels to belong to one and the same physical object. Thus, when stepping forward from the Stixel to the object level, the knowledge which Stixel belongs to which object is of particular interest, *e.g*. for collision avoidance and path planning.

For this purpose, we rely on the segmentation approach presented in [8]. Besides demanding motion consistency for all Stixels representing the same object, this scheme also makes strong use of spatial and shape constraints. The segmentation result for the depicted scenario is given in Figure 6d.

4.3. Pedestrian Recognition

Given our focus on urban scenarios, pedestrians and bicyclists are undeniably among the most endangered traffic participants. Rather than implicitly addressing pedestrian recognition solely as a generic object recognition problem using the stereo environment model sketched above, we additionally utilize an explicit pedestrian detection system in the near-range of up to 40 m distance from the vehicle. In doing so, we can exploit class-specific (pedestrian) models and obtain sufficient robustness for an automatic emergency braking maneuver.

Our real-time vision-based pedestrian detection system consists of three main modules: region-of-interest (ROI) generation, pedestrian classification and tracking. All system modules make use of two orthogonal image modalities extracted from stereo vision, *i.e*. gray-level image intensity and dense stereo disparity.

ROI Generation Adopting the approach of [14], ROI generation first involves the recovery of scene geometry in terms of camera parameters and 3D road profile from dense stereo vision. The current scene geometry constraints possible pedestrian locations regarding the estimated ground plane location, 3D position and height above ground. ROIs are then computed in a sliding-window fashion.

Pedestrian Classification Each ROI from the previous system stage is classified by powerful multi-cue pedestrian classifiers. Here, we are using a Mixture-of-Experts scheme that operates on a diverse set of image features and modalities inspired by [4]. In particular, we couple gradient-based features such as histograms of oriented gradients (HoG) [1] with texture-based features such as local binary patterns (LBP) or local receptive fields (LRF) [26]. Furthermore, all features operate both on gray-level intensity as well as dense disparity images to fully exploit the orthogonal characteristics of both modalities [4], as shown in Figure 7. Classification is done using linear support vector machines. Multiple classifier responses at similar locations and scales are addressed by applying mean-shift-based non-maximum suppression to the individual detections, *e.g*. a variant of [27]. For classifier training, we use the public *Daimler Multi-Cue Pedestrian Classification Benchmark*, as introduced in [5].

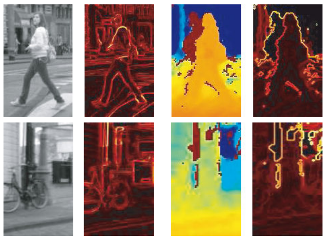


Figure 7: Intensity and depth images with corresponding gradient magnitude for pedestrian (top) and non-pedestrian (bottom) samples. Note the distinct features that are unique to each modality, *e.g*. the high-contrast pedestrian texture due to clothing in the gray-level image compared to the rather uniform disparity in the same region.

Tracking For tracking, we employ a rather standard recursive Bayesian formulation involving Extended Kalman Filters (EKF) with an underlying constant velocity model of dynamics. Pedestrians are modeled as a single point on the ground-plane. As such, the state vector holds lateral and longitudinal position as well as corresponding velocities. Measurements are derived from the footpoint of detected pedestrians and the corresponding depth measurements from stereo vision.

Pedestrians in areas to the side of the vehicle are particularly application-relevant in turning maneuvers. Given our limited field-of-view in the stereo system, see Figure 3, we additionally utilize a monocular variant of the pedestrian system described above, operating on the wide-angle camera that is also used for traffic light recognition.

4.4. Vehicle Detection and Tracking

Vision-based vehicle detection in the near-range (up to 40 m) involves a very similar system concept as is used for pedestrian recognition, see above. Additionally, we use the Stixel World as a compact medium-level representation to further narrow down the search space for possible vehicle locations, as suggested in [6].

However, the high velocities of approaching vehicles in relation to the autonomously driving ego-vehicle require a much larger operating range of our vehicle detection module than for pedestrian recognition. We consider vehicle detection and tracking at distances of up to 200 m from the ego-vehicle, see Figure 8. In such a long range scenario, precise depth and velocity estimation is very difficult due to large disparity noise, given our camera setup. For similar

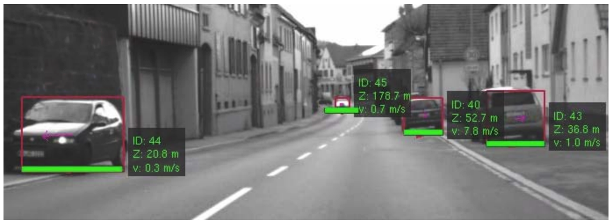


Figure 8: Full-range (0 m - 200 m) vehicle detection and tracking example in an urban scenario. Green bars indicate the detector confidence-level.

reasons, we cannot apply stereo-based ROI generation.

Thus, we rely on a very fast monocular vehicle detector in the long range, *i.e.* a Viola-Jones cascade detector [23]. Since its main purpose is to create regions-of-interest for our subsequent strong Mixture-of-Experts classifiers, as described above, we can easily tolerate the inferior detection performance of the Viola-Jones cascade framework compared to state-of-the-art and exploit its unrivaled speed.

Precise distance and velocity estimation of detected vehicles throughout the full distance range poses extreme demands on the accuracy of stereo matching as well as camera calibration. In order to obtain optimal disparity estimates, we perform an additional careful correlation analysis giving a sub-pixel accuracy of 0.1 px.

Moreover, we put a strong emphasis on the on-line calibration of the squint angle of our camera system to get precise distance estimates from the disparity map. Since we may assume that the long range radar of our vehicle delivers precise distance measurements, we run a slow disturbance observer to compensate for drifts of this very critical angle.

4.5. Traffic Light Recognition

Given the viewing angle of 45°, our stereo camera system is not well suited for traffic light recognition. Stopping at a European traffic light requires a viewing angle of up to 120° to be able to see the relevant light signal right in front of the vehicle. At the same time, a comfortable reaction to red traffic lights on rural roads calls for a high image resolution. For example, in case of approaching a traffic light at 70 km/h, the car should react at a distance of about 80 m which implies a first detection at about 100 m distance. In that case, given a resolution of 20 px/°, the illuminated part of the traffic light is about 2x2 px, which is the absolute minimum for successful classification. For practical reasons, we chose a 4 MPixel imager and a lens with a horizontal viewing angle of approximately 90°.

From an algorithmic point-of-view, traffic light recognition involves three main problems: detection, classification and selection of the relevant light at complex intersections. To avoid a strong dependency on the map, we apply an image based localization method consisting of an off-line and an on-line step, as follows.

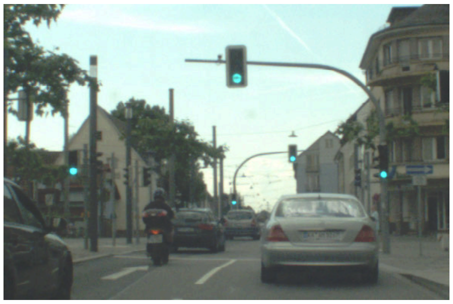


Figure 9: Example of an ideal situation with easy to detect traffic lights.

Off-line, an image sequence is recorded while driving towards the intersection of interest. For these recorded images, we compute highly discriminative features in manually labeled regions around the relevant traffic lights. These features are stored in a data base.

While driving in on-line mode, the features in the actual image are matched against this data base. The resulting matching hypotheses allow both the identification of the best matching image in the data base and the determination of the location of the relevant traffic light in the current image. The correspondent image regions serve as input for the subsequent classification step. Classification follows the principle introduced in [16]. The detected regions of interest are cropped and classified by means of a Neural Network classifier. Each classified traffic light is then tracked over time to improve the reliability of the interpretation.

The classification task turned out to be more complex than expected. While roughly $2/3$ of the 155 lights along the route were as clearly visible as shown in Figure 9, the rest turned out to be very hard to recognize. Some examples are shown in Figure 10. Red lights in particular are very challenging due to their lower brightness. One reason for this bad visibility is the strong directional characteristic of the lights. While lights above the road are well visible at larger distances, they become invisible when getting closer. Even the lights on the right side, that one should concen-

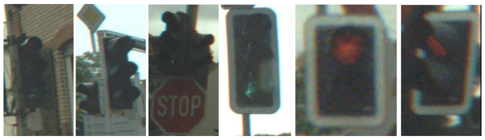


Figure 10: Examples of hard to recognize traffic lights. Note, that these examples do not even represent the worst visibility conditions.

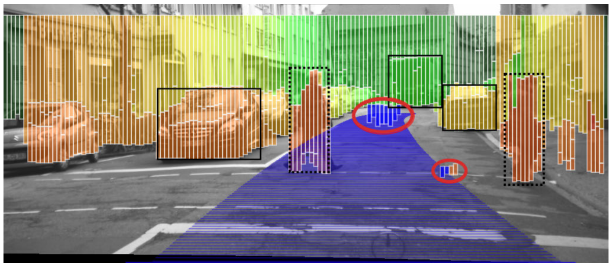


Figure 11: Visualization of our testing environment showing an example of an identified failure case. The Stixel-World (colors encode distance), the driven path (in blue) and recognized cars and pedestrians (black boxes) are shown. Due to an insufficient on-line calibration, small obstacles wrongly show up (red circles) in this scenario.

trate on when getting closer, can become nearly invisible in case of a direct stop at a red light. In those cases, change detection is more efficient than classification to recognize the switching between red and green. To improve visibility, we also decided to adjust the stopping position of the car in the global map, in case of a detected red light.

5. System Test and Validation

In the algorithm development and real-world testing phase, many people contributed to the vision system and continuously updated their software. To minimize the risk of serious software bugs and unexpected performance decrease, we built a powerful testing tool-chain following the established principles of unit testing, integration testing and system testing. To verify that new algorithmic releases meet their requirements, they were tested on hours of recorded sensor data of the route and had to show at least the same performance as the previous versions. Here, both the individual components as well as the fully-integrated system were put through their paces.

Since manual labeling is infeasible for such a large amount of data, we adopted the idea presented in [21] that allows for a semi-automatic object-level performance evaluation. Since the exact driven path is known, we can check the driving corridor for obstacles detected by our vision modules. We can safely assume that there are no static obstacles blocking our path up to two seconds in advance and that moving obstacles coincide with the radar-objects. An example of our testing toolchain is given in Figure 11.

During processing of the data base, statistics are generated and ambiguous situations reported. Later, a human observer can easily inspect situations where unexpected objects were detected. A helpful statistic is the time-to-collision (TTC) histogram shown in Figure 12. It depicts the likelihood that Stixels occur in our driving path at a certain TTC. Most of the remaining obstacles closer than 2 seconds are caused by newly initialized Stixels for which a reliable motion estimate is not yet available. Lower probabilities at small TTCs translate to a lower false positive rate. This allows us to use such statistics not only for verifying new releases but also for optimization of the different parts of the whole stereo analysis chain.

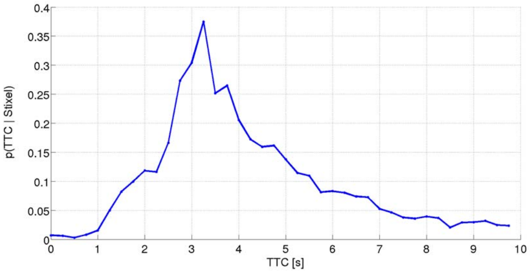


Figure 12: TTC statistics indicating the likelihood that Stixels occur at a certain TTC.

6. Results

During our tests, about 6,700 km were driven in fully autonomous mode. The final journey took place in August 2013 in busy traffic. The maximum speed in cities was 50 km/h, while on county roads the maximum allowed speed of up to 100 km/h was driven at most times. 54 km of the route are urban, 50 km are rural roads. There are no highways along the route. The route was driven in intervals, following an induced safety requirement of not more than 45 minutes for the control engineer behind the wheel. The total time required for the trip was about 3 hours. Bertha was able to handle all occurring situations including 18 busy roundabouts, numerous pedestrians and bicyclists on the road, 24 merge situations, as well as narrow passages in small villages, where parked cars forced the car to wait for oncoming traffic. No sudden human intervention was necessary. In total, the car asked for manual control twice when it stopped safely in front of an obstacle and did not see a chance to proceed. In the first situation, the lane was blocked by a construction site, in the second case, Bertha had stopped behind a van. Since we prohibited the car from entering the opposite lane by more than one meter, Bertha had no choice but to hand control back to the control driver.

Many people have compared Bertha to a human ”learner” taking driving lessons. Sometimes the car behaved extremely carefully, while in other situations an experienced human driver would have driven more defensively. However, we believe that - like a human - Bertha will improve its driving skills over time, which mainly translates to an improvement of the vision system.

7. Conclusions

Bertha successfully drove approximately 100 km of the *Bertha Benz Memorial Route* in public traffic in a fully autonomous manner. Besides the used radar sensors, machine vision thereby played the most significant role. It was indispensable for lane recognition, traffic light recognition, object recognition, precise free-space analysis, and object size and pose estimation. Although we optimized our vision algorithms for the route, its considerable length, complexity, dynamics and unpredictability coupled with the large variety of situations guarantees that we did not adapt too much to this particular road section. One of our goals was to identify the most important topics for further research:

Improve intention recognition This implies turn light recognition of leading or oncoming cars, the intention of pedestrians on the road, and above all the intention of bicyclists. Although reliably recognized by the pedestrian classification module and the radar, our safety driver felt particularly uncomfortable when passing a bicyclist.

Increase robustness During the development phase we encountered all possible weather situations including heavy snowfall, strong rain, and low sun. While light rain did not cause problems, snow on the street significantly impacted the lane recognition module. During strong rain, the disturbances on the wind shield caused problems for disparity estimation and hence to the whole subsequent module chain. We found that proper confidence measures for all reported objects are an absolute necessity.

Generate redundancy Stereo vision was used to analyze the situation in front of the car and monocular vision to detect traffic lights and pedestrians in turning scenarios. Automation requires redundancy and hence more cameras would be desirable.

Improve the imager For traffic light recognition, we specifically selected a CCD-imager since it showed better quality than a comparable CMOS sensor. Still, we desire better color quality and an automotive compliant CMOS sensor for increased dynamics.

References

[1] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *Proc. CVPR*, pages 886–893, 2005. 5

[2] E. D. Dickmanns. Vehicles capable of dynamic vision: a new breed of technical beings? *Artificial Intelligence*, 103(1-2):49–76, 1998. 2

[3] E. D. Dickmanns, B. Mysliwetz, and T. Christians. An integrated spatio-temporal approach to automatic visual guidance of autonomous vehicles. *IEEE Trans. on Systems, Man, and Cybernetics*, 20(6):1273–1284, 1990. 2

[4] M. Enzweiler and D. M. Gavrila. A multi-level Mixture-of-Experts framework for pedestrian classification. *IEEE Trans. on IP.*, 20(10):2967–2979, 2011. 5

[5] M. Enzweiler et al. Multi-Cue pedestrian classification with partial occlusion handling. *Proc. CVPR*, 2010. 5

[6] M. Enzweiler et al. Efficient stixel-based object recognition. *IEEE IV Symp.*, pages 1066–1071, 2012. 5

[7] M. Enzweiler et al. Towards multi-cue urban curb recognition. *IEEE IV Symp.*, 2013. 3

[8] F. Erbs, B. Schwarz, and U. Franke. Stixmentation - Probabilistic stixel based traffic scene labeling. *Proc. BMVC*, 2012. 5

[9] U. Franke and A. Ismail. Recognition of bus stops through computer vision. *IEEE IV Symp.*, 2003. 3

[10] U. Franke et al. Autonomous driving goes downtown. *IEEE Int. Sys.*, 13(6):40–48, 1995. 2

[11] S. Gehrig, F. Eberli, and T. Meyer. A real-time low-power stereo vision engine using semi-global matching. In *Proc. ICVS*, 2009. 4

[12] H. Hirschmüller. Accurate and efficient stereo processing by semi-global matching and mutual information. In *Proc. CVPR*, pages 807–814, 2005. 4

[13] T. Jochem and D. Pomerleau. Life in the fast lane: The evolution of an adaptive vehicle control system. *AI Magazine*, 17(2):11–50, 1996. 2

[14] C. Keller et al. The benefits of dense stereo for pedestrian recognition. *IEEE ITS*, 12(4):1096–1106, 2011. 5

[15] H. Lategahn et al. Urban localization with camera and inertial measurement unit. *IEEE IV Symp.*, 2013. 4

[16] F. Lindner, U. Kressel, and S. Kaelberer. Robust recognition of traffic signals. *IEEE IV Symp.*, 2004. 6

[17] No Hands Across America Webpage. www.cs.cmu.edu/afs/cs/usr/tjochem/www/nhaa/nhaa_home_page.html, 29 August 2013. 2

[18] D. Pfeiffer and U. Franke. Efficient representation of traffic scenes by means of dynamic stixels. *IEEE IV Symp.*, 2010. 4

[19] D. Pfeiffer and U. Franke. Towards a global optimal multi-layer stixel representation of dense 3d data. *Proc. BMVC*, 2011. 4

[20] O. Pink and C. Stiller. Automated map generation from aerial images for precise vehicle localization. *IEEE ITSC*, 2010. 3

[21] N. Schneider et al. An evaluation framework for stereo-based driver assistance. *Springer LNCS No. 7474*, pages 27–51, 2012. 7

[22] M. Schreiber, C. Knöppel, and U. Franke. LaneLoc: Lane marking based localization using highly accurate maps. *IEEE IV Symp.*, 2013. 3

[23] P. Viola and M. Jones. Robust real-time object detection. *IJCV*, 57(2):137–154, 2001. 6

[24] VisLab PROUD-Car Test 2013 Webpage. vislab.it/proud/, 29 August 2013. 2

[25] Wikipedia, Autonomous Car. en.wikipedia.org/wiki/Autonomous_car, 29 August 2013. 1

[26] C. Wöhler and J. K. Anlauf. An adaptable time-delay neural-network algorithm for image sequence analysis. *IEEE Transactions on Neural Networks*, 10(6):1531–1536, 1999. 5

[27] C. Wojek, S. Walk, and B. Schiele. Multi-cue onboard pedestrian detection. *Proc. CVPR*, 2009. 5