PROUD—Public Road Urban Driverless-Car Test

Alberto Broggi, Pietro Cerri, Stefano Debattisti, Maria Chiara Laghi, Paolo Medici, Daniele Molinari, Matteo Panciroli, and Antonio Prioletti

Abstract—**This paper presents an autonomous driving test held in Parma on urban roads and freeways open to regular traffic. During this test, the vehicle not only performed simple maneuvers, but it had to cope with complex driving scenarios as well, including roundabouts, junctions, pedestrian crossings, freeway junctions, and traffic lights. The test demonstrated the ability of the current technology to manage real situations and not only the well-structured and predictable ones. A comparison of milestones, challenges, and key results in autonomous driving is presented to highlight the novelty and the specific purpose of the test. The whole system is described: the vehicle; the software architecture; details about high-, medium-, and low-level control; and details about perception algorithms. A conclusion highlights the achieved results and draws possible directions for future development.**

Index Terms—**Autonomous vehicles, intelligent systems, image processing, data fusion, system architecture.**

I. Introduction

Autonomous driving has become a fast expanding and promising area: the development of autonomous vehicles has been speeding up in the last few years, thanks to the direct and increasing interest of automotive companies. VisLab has been active on this field for 20 years and marked many milestones, demonstrating the feasibility of new challenges with ever increasing complexity.

This paper presents a new test of autonomous driving on urban roads and freeways in Parma, where the vehicle had to negotiate, among other scenarios, two-way narrow rural roads, pedestrian crossings, traffic lights, roundabouts and freeway junctions. The test was a demonstration of the capabilities of VisLab autonomous vehicle to drive in real traffic.

Autonomous vehicles tests can be described according to different characteristics such as the use of digital maps, the availability of a prior knowledge on the scenario, the kind of environment (urban, highway, off-road, ...). A critical aspect is represented by obstacles: static obstacles, that can also be mapped, are quite straightforward to manage, but robust perception algorithms are mandatory in case of dynamic obstacles. If obstacles behavior can not be predicted, perception has to be very sharp and the path planner must also consider this issue. Moreover, complexity grows with speed, as well: 40–50 km/h can be considered the threshold between slow and fast speeds.

Since the 1990s, different tests and challenges have been conducted to demonstrate and compare autonomous vehicles capabilities. In the following, different taxonomies will be surveyed, starting from platooning and open challenges. In August 1997, The US National Automated Highway System Consortium (NAHSC) presented a proof of technical feasibility of automated driving. It took place in San Diego, California [1], [2] and demonstrated many different functions such as vision-based lane keeping, magnetic nails following, platooning, cooperative maneuvering, and obstacle detection and avoidance. Within the platoon, vehicles were coordinated closely with through communication link, allowing each vehicle to exchange data with neighboring vehicles and roadside infrastructure.

Autonomous platooning was the main theme of the 2011 Grand Cooperative Driving Challenge (GCDC), providing the possibility for teams to develop and compare their cooperative driving solutions in a competitive setting [3]. The challenge consisted of two platooning scenarios, urban and highway, and was mainly focused on the ability to perform longitudinal control of the vehicles. In 2016 a new edition of GCDC [4] will require the addition of lateral control (steering) and participants will have to demonstrate the ability to merge platoons and to join a busy road on a T intersection without driver intervention.

Third-party tests played an important role and formulated the related technical indicators to guide the development of autonomous ground vehicles [5]. In particular the Grand Challenges [6], [7], sponsored by DARPA in 2004 and 2005, pushed the capabilities of autonomous vehicles to a new level, requiring the completion of lengthy and difficult off-road course across desert southwest areas of the United States, navigating at speeds between 5 and 50 mph and without any manual intervention. The vehicles had to follow a single path defined by GNSS (Global Navigation Satellite System) coordinates, navigating within the race corridor as well as negotiate or avoid other vehicles, fences, utility poles, stones, trees and bushes, ditches, and other natural or man-made static obstacles within the route corridor.

In 2007 DARPA conducted a third competition, the Urban Challenge [8], that featured autonomous vehicles driving a complex course in a staged city environment, negotiating other moving traffic and obstacles while obeying traffic regulations. This event was groundbreaking as the first time autonomous vehicles have interacted with both manned and unmanned vehicle traffic in an urban environment.

In China, the first third party test for the autonomous ground vehicle competition, named the Intelligent Vehicle Future Challenge, was held in 2009 with the support of the National Natural Science Foundation of China. Other competitions have been held every year with well designed test and evaluation

Manuscript received March 27, 2015; revised June 19, 2015; accepted August 29, 2015. Date of publication September 23, 2015; date of current version November 23, 2015. The Associate Editor for this paper was P. Zingaretti.

The authors are with VisLab, Dipartimento di Ingegneria dell'Informazione, Università degli Studi di Parma, Parma 43100, Italy (e-mail: broggi@vislab.it; cerri@vislab.it; deba@vislab.it; laghi@vislab.it; medici@vislab.it; dmolina@vislab.it; panci@vislab.it; prio@vislab.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2015.2477556

systems, with the aim of achieving autonomous driving in real traffic. IVFC 2009 and IVFC 2010 took place in constrained roads, while IVFC 2011 in open urban roads, with slight artificial modifications. IVFC 2012 was a demonstration of fully autonomous driving on freeways, including lane changing and traffic following, and in 2013 the competition focused on the so called 4S (i.e., safety, smartness, smoothness, and speed) [9] putting vehicles through both suburban and urban road sets, and requiring to slow-down in front of a school and stopping for pedestrians.

Unlike the unmanned vehicles developed in other countries, which basically rely on GNSS information and navigation of electronic maps, the IVFC requires the participating vehicles to perceive the environment with the equipped sensors such as vehicle-mounted cameras, radar, and lidar sensors. The 2010 competition program [10] required that vehicles were navigated only by machine vision and laser range finders since GPS or any other satellite navigation system was forbidden by regulation.

In the 2011 competition, static and moving vehicles were introduced to test the competitors ability to interact with traffic [5]. The track was kept confidential, and vehicles had to work in an environment with high uncertainty depending on their on-board devices and self-driving system. The competition contents included traffic sign recognition, obstacle avoidance, merging into traffic, U-turn, intersection, and lane changing. GNSS was not forbidden, but the test task was designed such that vehicles could not finish the race only relying on it.

In the Hyundai Autonomous Challenge [11], that took place in Seoul in November 2010, contestants were required to build an autonomous vehicle capable of following a course and performing missions like avoiding obstacles, passing through gates and tunnels, stopping in front of a pedestrian crossing or a stop line, and slowing down in front of speed bumps. Such a challenge explicitly required the use of vision to detect patterns. Other two challenges were held in 2012 and 2014, increasing the difficulty of proposed missions.

The development of an autonomous vehicle architecture is a complex problem, differently approached by research groups in the realization of their prototypes.

During MilleMiglia in Automatico [12], in 1998, ARGO vehicle drove itself for more than 2000 km in automatic mode on Italian highways, passing through flat areas and hilly regions including viaducts and tunnels. ARGO used only passive sensors (cameras and a speedometer) to sense the surrounding environment.

With the "Stadtpilot"-Project [13], [14], the Technische Universitat Braunschweig transferred the knowledge gained from its participation in the DARPA Urban Challenge to the real urban environment of Braunschweig's inner ring road. The goal was to drive fully autonomously in the traffic flow and to behave according to traffic rules. On October 8th, 2010, the "Stadtpilot"-vehicle conducted challenging autonomous driving maneuvers, such as lane keeping, intersections joining, obstacles avoidance, and control speed according to the flowing traffic. The demonstration took place in urban traffic, and the maximum speed was about 60 km/h. To improve both accuracy and integrity of the positioning information, a landmark-based positioning system was used.

Google [15] extended experience gained in the Urban Challenge by Standford University, with an autonomous car that can drive fast on familiar and well mapped road, managing any kind of obstacle. A roof-mounted high-end laser scanner and a detailed map, recorded in a prior manual drive, provide the main information about the driving environment.

Google car's early life was confined almost entirely to California highways. After hundreds of thousands of test miles Google's team shifted focus from predictable environments, such as freeways, to the unpredictable maze of city streets. The vehicle strongly relies on the use of digital maps, since Google researchers consider them one of the key insights that emerged from the DARPA challenges. Maps give the car a baseline expectation of its environment; moreover the car collects sensor data from its radar, lasers, and cameras, and can detect hundreds of distinct objects simultaneously.

In August 2013, an S-Class vehicle with close-to-production sensors drove completely autonomously for about 100 km, following the well-known historic Bertha Benz Memorial Route [16], [17]. The autonomous vehicle handled traffic lights, pedestrian crossings, intersections, and roundabouts in real traffic. It had to react on a variety of objects including parked cars, preceding and oncoming vehicles, bicycles, pedestrians and trams, thus testing the employed vision algorithms for object recognition and tracking, free-space analysis, traffic light recognition, lane recognition, as well as self-localization in a non-structured and unpredictable environment. An important source of information for Bertha vehicle is a detailed digital map containing the position of lanes, the topology between them as well as attributes and relations defining traffic regulations (e.g. right-of-way, relevant traffic lights, and speed limits). A prerequisite for using such digital maps is a precise map-relative localization.

In June 2014, the final demonstration of the VCharge (Automated Valet Parking and Charging for e-Mobility) EU project has been presented [18]. In this test an electric autonomous car has to move at low speeds in a parking area, managing static and dynamic obstacles, find free parking lots, stop and restart at intersections, and reach its recharging station in case of low batteries. The vehicle only uses low-cost and close to market sensors.

In Table I, all the described tests and challenges are summarized, and comparative information about approaches, sensors, and scenarios is provided.

The underling idea of PROUD test was to engage a new challenge, i.e., autonomous driving in open public roads. During VIAC (VisLab Intercontinental Autonomous Challenge) [19], the vehicles were able to drive in unmapped and unknown scenarios, managing any kind of obstacles, at low speeds. Due to the lack of digital maps a leader-follower approach has been used to manage most of the trip. When the leader is not visible by the follower, the second vehicle follows the coarse GPS waypoints broadcasted via radio connection by the first vehicle or keeps the lane using information provided by a vision-based lane markings detection system. The idea of PROUD is to move the VIAC experience to a different level: moving in a mapped and familiar scenario, it is possible to drive faster and to trigger the correct driving mode. An openly licensed map

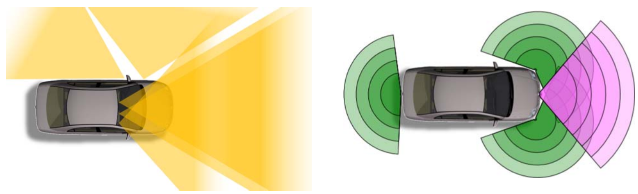
TABLE I
INTELLIGENT VEHICLES TESTS

Type	Test	Scenario	Vehicles	Sensors	Approach
Platooning	NAHSC 1997	Urban and inter-city highways	Fully autonomous cars and city buses, trucks	Vision systems and radar, buried magnetic markers	Free agents and cooperative driving, platooning, vision-based road following
	GCDC 2011	Urban and highway platooning	Cars and trucks	Radar, lidar and vision systems	Cooperative adaptive cruise control, V2x communication
Third-party competitions	DARPA Grand Challenges 2004, 2005	Unstructured off-road course up to 175-miles long in less than 10 hours	Autonomous cars and trucks	Lidars, GNSS/INS, Cameras	Waypoint following, negotiate or avoid other vehicles and obstacles, no manual intervention allowed
	DARPA Urban Challenge 2007	Complex course in a staged city environment	Autonomous cars and trucks	Cameras, lidars, GNSS/INS	Mission based, negotiating other moving traffic and obstacles while obeying traffic regulations
	IVFC 2009-2010	Constrained roads	Autonomous cars	Cameras and lidars	Navigation, recognition of traffic signs, S-load running, parking. Use of GNSS was forbidden
	IVFC 2011	Open urban road, with slight artificial modifications	Autonomous cars	Cameras, radar and lidar sensors, GNSS/INS	Environment perception, navigation, decision making and automatic driving; human intervention is not allowed
	IVFC 2012	Beijing-Tianjing Highway (114 km)	Autonomous cars	Cameras, radar, GNSS/INS	Fully autonomous driving without human intervention, cruising in one lane, following traffic, changing lanes, and responding to human instructions
	IVFC 2013	Suburban and urban road tests (23 km)	Autonomous cars	Cameras, radar and lidar sensors, GNSS/INS	Natural environment perception, 4s scale (safety, smartness, smoothness, and speed)
	Hyundai Autonomous Challenge 2010	Constrained path	Autonomous cars	Cameras, lidars, GNSS/INS	Follow a course and perform missions avoiding obstacles; vision systems are required to detect patterns
Self-organized tests	MilleMiglia in Automatico 1998	2000 km on Italian highways	Autonomous vehicle equipped with vision systems and automatic steering capability	Passive sensors only (cameras and speedometer)	Determine own position with respect to the lane, compute the road geometry, detect generic obstacles on the path, localize a leading vehicle
	Braunschweig Stadtpilot-Project 2010	Urban environment of Braunschweigs inner ring road	Autonomous car vehicle	GNSS/INS, lidars	Lane keeping, interaction with traffic, lane change maneuvers at speeds up to 60 km/h
	Pikes Peak 2010	Extreme off-road	Autonomous car	GNSS, INS	Waypoint following at high speed
	Google autonomous car tests	Highways and city streets	Autonomous car	64-plane lidar, radar, and camera	Map based navigation, lane and speed keeping, lidar-based dynamic objects detection
	Bertha Benz Memorial Route 2013	Overland passages, urban areas, small villages (103 km)	Autonomous car equipped with close-to-market sensors	Radar, monocular cameras, stereo cameras	Stereo vision, self-localization, free-space analysis, object detection
	VIAC 2010	Real traffic roads from Italy to China	Four autonomous electric vans	Monocular cameras, stereo cameras, GNSS/INS, lidars	Vehicle following (virtual towing), no digital maps are used
	PROUD 2013	Real traffic roads in freeway and urban scenarios	Autonomous car	Monocular cameras, stereo cameras, GNSS/INS, lidars	Waypoint following, Lane keeping, Obstacle avoidance, roundabouts, junctions, pedestrian crossings, traffic lights

has been enriched with information about the maneuver to be managed (e.g. pedestrian crossing, traffic light, ...) and also the positioning has been refined to reach a lower error rate, but there is no a priori knowledge about static obstacles: perception of the world was totally made in real time. The PROUD test demonstrates that it is not necessary to use precise 3D maps: world reconstruction, including both road features and road users, can be performed during the driving.

II. The BRAiVE Vehicle

To perform the PROUD test, the BRAiVE prototype has been slightly modified starting from the configuration already presented in [20]. The idea was to improve the perception systems, focusing on weaknesses of the previous platform. The sensor suite is designed addressing to the specific issues that the prototype must deal with: for this reason, for example, the right side area of the vehicle is covered only by a laser scanner.



(a) (b)

Fig. 1. Sensors used during the PROUD test. (a) Cameras and (b) lasers coverage.

In particular two Hokuyo UTM-30LX-EW laser scanners were mounted on the front of the vehicle, to replace other lower performance devices. The positioning system was also improved: a Topcon AGI3 with real-time RTK correction by GEOTOP with a new external antenna was installed. Extra care was paid to data synchronization: the IBEO Lux output trigger signal has been used to synchronize all the cameras.

All devices were partitioned and connected to different PCs with the following rationale:

- sensors framing the same world area are managed by the same PC.
- the computational load is equally distributed among all PCs.

Three PCs were devoted to perception:

1) Frontal obstacle and lane detection using a frontal stereo camera and a IBEO 4-layers laser scanner;
2) Lateral obstacle detection using two single layer laser scanners and two cameras mounted in front bumper corners;
3) Rear and approaching obstacles detection plus road markings (pedestrian crossings, stop lines), and traffic lights using single-layer laser scanner mounted in the rear bumper, cameras mounted in the external rearview mirrors facing backwards; an additional frontal camera is connected to this PC.

Another computer was used for vehicle control: high, medium, and low level control plus a component called World Perception Server (WPS). This computer was also connected via CAN Bus to the dSpace MicroAutoBox, in charge of communicating with the vehicle Electronic Central Units.

Figs. 1 and 2 show sensors used for each application.

III. System Architecture

The specific architecture built for the PROUD test is the final result of the "Open intelligent systems for Future Autonomous Vehicles" (OFAV) project, and aims at creating a fluid, modular, and scalable system, with the use of only open source data, especially regarding the navigation maps.

The architecture is composed of three main layers: perception, planning, and control; the *World Perception Server* (WPS) performs multilevel data fusion and provides data to the planning layer, also acting as a separator between the perception and planning layers. The WPS is described in detail later in this section, together with some of the perception applications.

Since each architecture component has been developed as an independent application, a message passing service has been developed to perform inter-process communication, thus achieving modularity and scalability. This component is an abstraction of a physical bus that models the inter-process communication as a data bus: several data buses are available, whose lines represent different communication topics. This Data Distribution Service allows information exchange between processes on the same computer(using shared memory) or between different computers, across the network (via UDP/IP multicast packets), in a transparent and efficient way. A similar system is presented in [21].

All the perception applications running during PROUD have been developed into the VisLab's framework called *GOLD* [22], while both WPS and the modules of the planning layer are standalone applications.

Fig. 2 shows the whole system architecture.

Due to the great complexity of the automotive scenario, especially regarding the planning and decision making phases, a hierarchical approach is adopted, splitting the planning layer in three sub-layers: Navigation layer, Maneuver layer, and Control layer [23]: in this paper these layers will be also referred to as high, medium, and low level control. The Navigation and Maneuver layers are explained in Section III-C, while the Control layer is presented in Section III-D.

A. Perception

Several perception systems were integrated to create a realistic world representation. In order to have a 360 degree world view, each algorithm has been executed using data coming from different sensors around the vehicle, as described in Section II. In the following all the perception algorithms are described.

1) Obstacles Detection: Two kind of obstacle detectors have been used: the Vision-based Obstacle Detector (VOD) described in [24] and the Laser-based Obstacle Detector (LOD).

The VOD is composed by a stereo-based 3D world reconstruction, an obstacle clustering which uses a occupancy grid map, and a tracking stage [25].

The LOD uses a distance-based clusterization function to represent obstacles; their shapes are estimated merging laser echoes into a polyline.

Inter-frame association is based on the distance between obstacles centroids.

The output representation of obstacles contains the shape, the distance from the vehicle, and the distance uncertainty.

Filtering process for both applications is presented in the World Perception Server section. Rear-view mirrors cameras are used to detect oncoming traffic when merging into the highway. The system uses both a classification algorithm (based on Haar feature and AdaBoost) and a feature tracking algorithm that highlights moving obstacles. A similar algorithm is also developed for vehicle detection while merging a junction [26], but during the final test it was not enabled because it was not fully tested.

2) Lane Detection: Lane markings detection is performed extracting the Dark-Light-Dark patterns on the bird-eye view image. Solid, dashed, single, and double lines are detected.

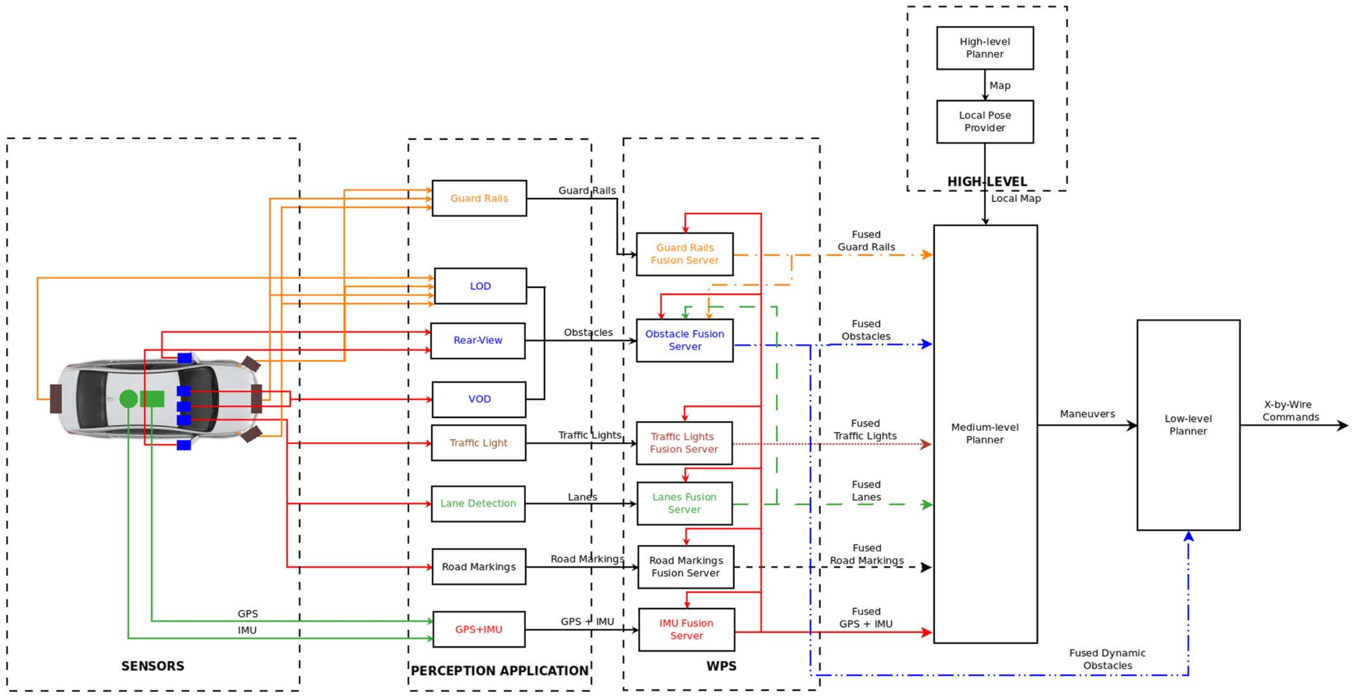


Fig. 2. Scheme of the whole system.

A detailed description of the algorithm can be found in [27]. Lane markings are represented as polylines with uncertainty associated to each point. Detection is performed on images coming from a frontal camera and from rear-view mirrors cameras.

3) Other Road Markings Detection: The detection is divided in two phases:

- the extraction of bright regions through the Maximally Stable Extremal Region (MSER) algorithm;
- a high level stage that compares detected regions with the expected shape, depending on searched road marking type.

Road markings (pedestrian crossing, stop line, highway junction) are represented as segments.

4) Guard Rails Detection: Guard rails and generic barriers are searched among obstacles provided by LOD. Obstacle points are used to fit a line or a parabola that could describe a guard rail. The output, as in the lanes detection application, is a list of polylines with uncertainty associated to each point.

5) Traffic Lights Detection: A color based clustering has been used to detect traffic lights. Moreover, by knowing camera parameters and the minimum and maximum height of traffic lights according to Italian laws, search area is reduced in order to speed up the system and remove false positives. The system is described in [28]. Traffic lights are represented by their position and their state (green, red, yellow).

B. World Perception Server

The World Perception Server (WPS) is responsible for receiving data from the perception algorithms and fusing them in order to create a unique global perception map.

The registration to the server is handled by a two-way handshaking process; two types of users are managed: the suppliers (perception algorithms) and the clients (planners). To prevent from wrong object tracking and data estimation, the WPS checks the synchronization time on each request, comparing its time stamp with the last provided data time stamp: the WPS can detect desynchronization at the first request and rejects data coming from the supplier.

1) Sensor-Side: Perception application registration is divided in two phases:

- in the first one, the application sends information to the server regarding the type of output that is being provided, the presence of tracking information, sensor position, and field-of-view;
- in the second one, the WPS replies to the application whether it can manage the perception type. In case of positive response, the server provides an ID of the channel where the application will send its data; otherwise the answer can be a “refused” message or, if the WPS is not ready, a “try later” message.

Fusion servers, which have the task of managing the different perception types, are shown in Fig. 2.

Since the WPS is a time based application, it requires each object reported from all its registered applications to be timestamped. An NTP server is used to sync all the PCs and reach a synchronization level with a relative error less than 1 ms, enough for our purpose. To prevent WPS from waiting for a damaged sensor or a blocked application, the registered perception algorithms must send a heart-beat message to the server, otherwise they will be unregistered. Fusion service for a specific object is enabled if at least one sensor providing this kind of data is registered, otherwise the service is stopped and clients notified.

2) Planner-Side: Planner registration is divided in two phases, as well:

- a first phase, where the planner queries the WPS for its capabilities;
- a second phase, where the WPS provides a channel pair for each requested capability: one bidirectional to exchange command/information messages, and one unidirectional where the server provides the data.

Planner requests contain the time at which the perception map has to be referred to: the stored data are projected to represent the view at the desired time.

WPS can handle several data types, as shown in Fig. 2: inertial, obstacles, lanes, barriers, traffic lights, and road markings. A central fusion server receives planner requests that are then forwarded to the specific fusion server. Each single fusion server is described in the following.

1) Obstacles: An obstacle is represented by:

- a list of its 2D points projected on the ground;
- its state, static or moving, with an associated uncertainty that increases if remains in the same state;
- a bounding box, represented by the minimum box that contains all obstacle points;
- a lane-level localization.

Obstacles coming from the same perception algorithm are associated by the WPS only if the application does not support tracking; instead, obstacles coming from different applications are fused by the server. An association is valid if the minimum distance between obstacles bounding boxes is less than their maximum position uncertainty. Data are filtered through a Kalman filter, with the state represented by the obstacle centroid position and its speed. Due to the large difference between obstacles centroids coming from different kind of applications, it is not possible to fuse them in the same Kalman filter; therefore, a different Kalman filter has been used for each application and the fusion stage is performed after the tracking process.

In addition to the tracking cues, barrier fusion information is used to classify an obstacle as static or moving: if an obstacle is fused with a barrier it is always classified as static.

2) Lanes: Lane markings are represented by polylines. Fusion is divided in two levels: low-level fusion and high-level fusion. Association and merging of lane markings is performed in the low-level. Lane markings are reconstructed in order to have the same number of segments and to perform a segment-to-segment association: an average distance is obtained comparing each segment with the closest one of the other lane marking and, if their distance is under a certain threshold, the lane markings are merged.

Left, right, and own lane are computed in the high-level step; three steps can be distinguished:

- the medium points of the two lane markings are computed;
- a parabola is fitted using these points;
- a multi-frame filter is applied to stabilize the results and remove spikes for the navigation.

3) Barriers: Barriers fusion is based on the same algorithm of lanes fusion. The only difference is in the output: left and right barriers are computed fitting a parabola on the polylines points.

4) Traffic Lights: A traffic light is represented by its position and state. The tracking stage is based on a simple association performed using the position only. The output state is checked to respect the color order defined in Italian regulation (green, amber, red).

5) Road Markings: Pedestrian crossings, stop lines, and highway junctions are represented as road markings. The association between two road markings of the same type is performed with a minimum segment-to-segment distance metric.

6) IMU: This system provides two outputs: a precise relative motion of the vehicle and a rough global position localization. Relative vehicle motion is provided fusing 3 MEMS gyroscopes and accelerometers with precise wheels odometry; the localization uses also a GNSS with RTK correction. In case of low satellites number visibility condition, an estimation of vehicle position, based only on the other sensors, is provided. A Sigma Point Kalman Filter is used to merge all those data together.

C. Navigation and Maneuver

The high level control creates a path composed of a set of roads and junctions, connecting the current vehicle position to the destination specified by the user. Maps coming from the "Open Street Map Project" [29] have been used as a cartographic source, thus fulfilling the aforementioned constraint of openness of the used data.

To allow a faster path planning, these maps have been modified with the creation of a graph-based compressed map, the *Road Junction Map*. In this graph each edge is a road segment and each node represents a fork between two or more roads.

The *Road Junction Map* has approximately 40% less nodes than the corresponding OSM map, and allows to plan faster.

Route planning is based on a heuristic planner, derived from A* [30], where the cost function $c(x) = d(x_i, x_j)v(x_{i,j}) + \delta$ is based on the travel time of an edge plus a constant δ that has the aim to avoid unwanted path changes. The heuristic function of the planner is $h(x) = D(x, x_f)k_v$, which represents the theoretical travel time using the Euclidean distance and an average speed k_v.

A set of different maneuvers (e.g. lane keeping, vehicle following), grouped in high-level sets called *Driving Modes*, composes the Maneuver Layer. The goal of this layer is to create a traversability map for the low level path planner, choosing the right maneuver from the maneuver database and taking into account both road and traffic situations. Maneuvers, driving modes and the relationship between them are presented in [31].

Each Driving Mode (DM) is composed by many different maneuvers describing the various ways in which the target destination can be reached. In every DM each maneuver has a priority value and the trajectory planner of the low level control will select, in the current DM, the trajectory generated by the maneuver with the highest priority value. As an example, it is possible to configure the maneuver *Waypoint Following* with an higher priority value than the *Lane Keeping* maneuver, thus hiding the current perception data source. This technique has been extensively used during the PROUD demonstration to switch between *Waypoint, Lane Keeping* and other maneuvers in case of GPS outage or other possible temporary perception failure.

Each maneuver has a base structure composed by a list of pre-conditions, a list of post-conditions, a list of exit-conditions, and a list of gates. In particular, the concept of gate is the key idea of this level of planning: it integrates the speed, position and orientation that the vehicle must have in the future trajectory without directly planning an entire trajectory [31]. A gate is similar to a checkpoint planned in the future vehicle path, through which the vehicle must drive. Starting from the gate list, a dedicated component can create the traversability map needed by the Control Layer path planner. A key aspect of the medium level control is that it simply creates an ideal path that the vehicle must follow, without dealing with obstacles and with obstacle's trajectory, that are taken into account only by the low level control.

D. Path Planner

The path planner used by BRAiVE during PROUD has been implemented starting from several concepts developed for and after the VIAC experiment [32]. The planner controls both steering and speed in order to implement obstacle avoidance and collision mitigation functions directly in the same module.

To perform these complex tasks the planner requires the list of obstacles (static and dynamic) and the traversability area. These data are provided by the WPS (obstacles list) and Maneuver Level (traversability area). Starting from the static obstacles list and the traversability area, a potential map is generated, where low potential areas represent a good path to follow and high potential areas are sections of the world that it is better to avoid. Obstacles and the traversability area are merged together using specific blending functions in order to discourage trajectories that are too close to the area border or to the obstacles.

Trajectories are generated according to the motion equation

$$\begin{aligned} x' &= \cos\theta(s) \\ y' &= \sin\theta(s) \\ \theta' &= \kappa(s) \end{aligned} \tag{1}$$

where (x, y, θ) are position and orientation of vehicle, the derivatives are taken with respect to the covered distance s, and the curvature $\kappa(s)$ is a certain function described below. To generate trajectories with high smoothness, $\kappa(s)$ has been chosen belonging to the class of quadratic splines. Using the motion equation, feasible trajectories are generated in space coordinates as it is possible to see in Fig. 3.

These trajectories are evaluated on the map, measuring the potential along the vehicle perimeter and, in order to improve performances, only 7 points on the vehicle perimeter are considered during the evaluation. To further discriminate similar trajectories, a small correction factor is added to the score in order to take into account the final orientation of the vehicle and the maximum lateral curvature to improve comfort and

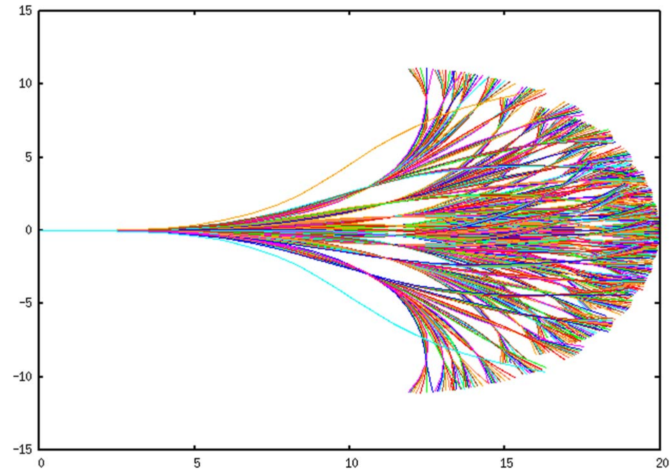


Fig. 3. Example of the precomputed trajectories.

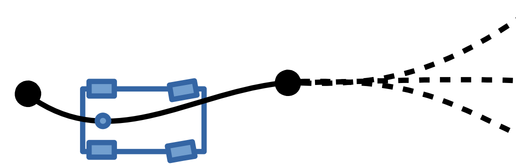


Fig. 4. Example of the construction of the trajectory: the first segment is kept constant.

safety feeling. The trajectory that collects the lower score is considered the best one and it is reported to the planner.

Moreover, to improve a bit the performance, the evaluation stage is subdivided in two subsequent steps.

Firstly, a rough subset of offline precomputed curves is evaluated on the map. Up to 6 GB of RAM are used to store the precomputed trajectories subdivided in 4 classes, quantized according to current speed (0–5 m/s, 5–10 m/s, 10–15 m/s, over 15 m/s), and indexed by the initial vehicle curvature. Each class of trajectories is characterized by a curvature limit computed from the associated speeds. For each of these classes, only trajectories that satisfy dynamic feasibility and comfort requirements on lateral acceleration are considered.

The best precomputed trajectory is then used as an initial estimation for the next non-linear optimization stage that enforce continuity with the previous maneuver and uses correct, and not quantized, initial vehicle curvature and speed.

One best trajectory is extracted at each planning stage, approximately any 80 msec. It is important to note that the number of spline knots for the curvature function $\kappa(s)$ is limited to 4 points and cannot represent any possible space trajectories: for this reason at each planning step there is no guarantee that the previous selected path, translated according to the vehicle movement, is included in the search space. To cope with this problem the first spline segment of the previous best trajectory is preserved and, if it proves to be obstacle-free, the new trajectory will be computed starting from this segment endpoint (see Fig. 4).

The final trajectory is a collision free path based on static obstacles and it is represented in space-domain. To generate the speed profile and deal with dynamic obstacles, a time-space collision map is generated and used. All the information

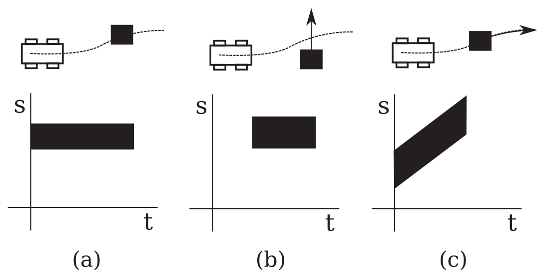


Fig. 5. Examples of the space-time map used to generate the speed profile: (a) Static obstacle. (b) Crossing obstacle. (c) Front vehicle.

provided by the system, such as the limit on the maximum lateral acceleration, the road speed limit, and the motion of dynamic obstacles, are used to generate this map. Example of these time-space maps are represented in Fig. 5.

Exploiting this map, a speed profile is computed in order to avoid or mitigate collision with dynamic obstacles, be consistent with road speed limits, and to achieve a smooth and comfortable driving experience. In order to increase the comfort experience two different speed limits are exploited: a soft limit (80% of speed limit, comfort lateral acceleration, safety follow front vehicle) and an hard limit (road speed limit, collision avoidance). A quantized set of speed profiles are evaluated and a score is assigned to each profile, considering final speed and which limit is violated.

Finally, the trajectory and speed profile are provided to the low level controller, where a Model Predictive Control (MPC) is used to convert space-domain $\kappa(s)$ to time-domain steering wheel set point $\delta(t)$ according to kinematic vehicle model. All the non-idealities of the system (inertia of steering wheel, vehicle, inertia and control delay of actuators) are represented as a black box and an ARX model is used to link input and output of the system. Since a new trajectory is provided each 80 msec, a MPC window of 400 msec is considered enough to deal with all the delay of the lateral control. Using the ARX model, the best quadratic spline is selected in order to minimize the difference between the requested trajectory and the simulated one.

Finally, to deal with different weather, road conditions, and vehicle set-up, vehicle parameters are evaluated online constantly, comparing the predicted path to the performed one.

IV. Test Results

The test was conducted in Parma from the University campus to the town center through different scenarios such as urban, rural, and highway roads; a roadmap is shown in Fig. 6. The test was conducted in open roads together with regular traffic at 11 am on a working day (Friday, July 12th 2013).

For security reason a human operator was sitting in the driver position to be able to control the car whether it would have been necessary or mandatory, but during the PROUD test no human intervention had been needed.

BRAiVE completed the whole track autonomously, along a 13 km route divided into 0.7 km of rural roads, 1.7 km of urban environments and 10.6 km of highways also outlined in Table II.

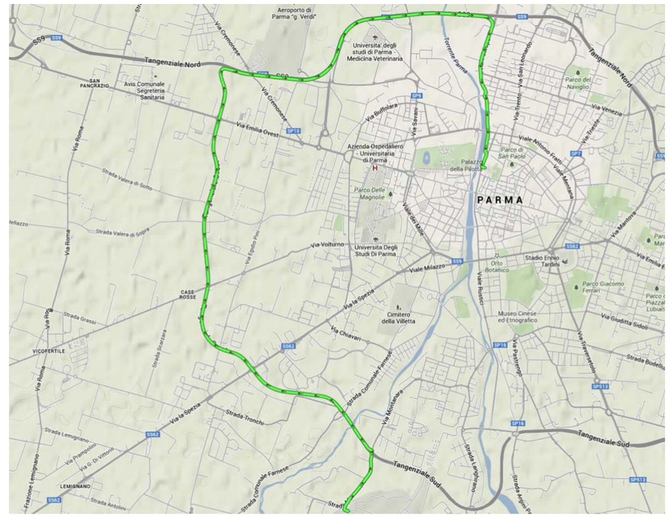


Fig. 6. Track overview with the vehicle's path superimposed on the map.

TABLE II
Path Statistics

Total Distance	13 km
Time of day	11:00 am
Day of week	friday, working day
Urban, suburban environments	2.4 km
Highway environment	10.6 km
Number of roundabouts	6
Number of junctions/merging	3
Number of crosswalks	15
Number of tunnels	8
Number of speed bumps	3
Number of traffic lights	1

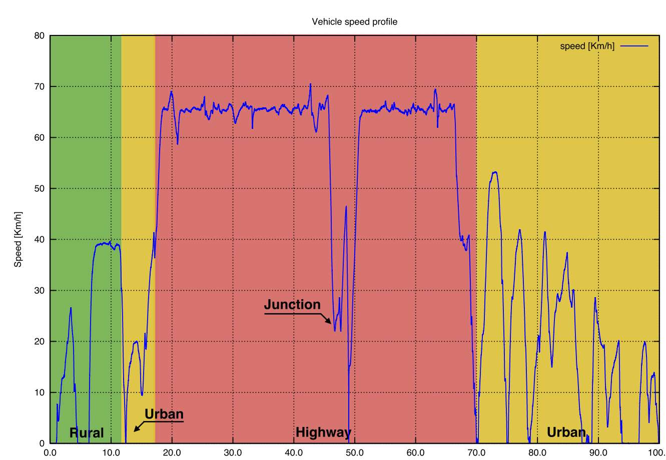


Fig. 7. Speed profile during the PROUD test.

Fig. 7 shows the speed of BRAiVE during the final test: on the freeway the speed was between 60 and 70 km/h, while on urban roads the speed was reduced, accordingly to the limits set in the map.

Fig. 8 shows the vehicle maneuvers that were engaged by the autonomous vehicle during the test; on the y-axis a list of the possible implemented maneuvers i.e.,: GNSS-based

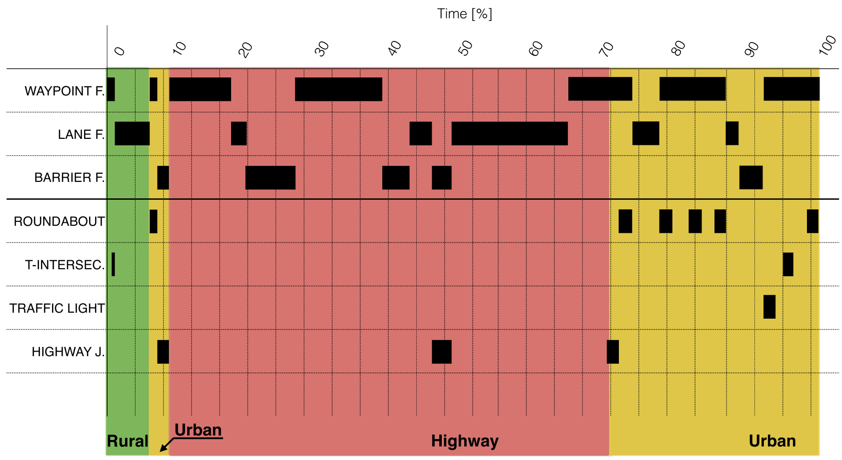


Fig. 8. Maneuvers overview, the environment is specified by the background color, i.e., green for rural, yellow for urban and red for highway.

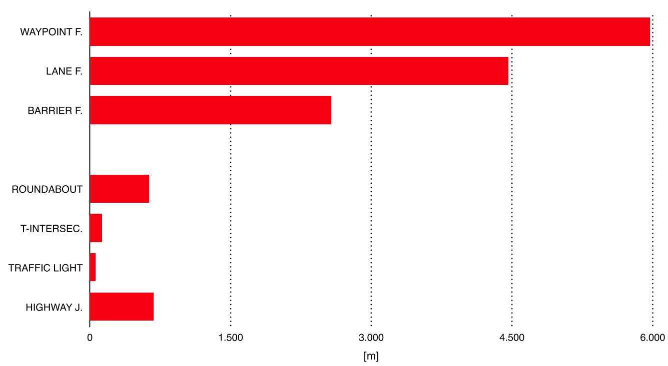


Fig. 9. Maneuvers usage.

Waypoint Following, Lane Following, Roundabout maneuver, T-Intersection maneuver, Traffic Light maneuver, and Highway Junction (those maneuvers are detailed in [31]).

Fig. 9 shows the distance driven performing each specific maneuver, thus highlighting which ones were mainly used. During all tests that have been made preparing the final demonstration, around 40–50% of the distance was covered without GNSS signal or with an imprecise localization. In such situations, other perception systems were activated to localize the vehicle and drive, e.g.: lane keeping and barrier detection. The medium level planner is in charge of replacing GNSS based localization with other localization systems. Also during the final demo, GNSS information has been used for less than 50% of the covered distance.

Table III shows the track statistics, highlighting also the GNSS performance.

GNSS outage occurs when the GNSS device runs either in standalone mode nor not enough satellites are available.

The perception algorithms were tested on demo track and qualitative results were extracted.

In the following these results are presented.

- Obstacle Detection. The proposed obstacle detection algorithms have been tested several times on the same path of the demonstration and they yielded qualitatively good results. Mainly, empirical tests have been conducted on recorded data by evaluating correct detection, tracking, and precision level (separation level). LOD implementation could provide better results than VOD in guard rail detection, thanks to its wider field of view. VOD reached good results even in case of strong vehicle pitch or road slope variations. VOD is based on a stereo pair with 40 cm baseline and a CCD camera with 6 mm lenses. The working range is 4m to 80m and, running on the described PC, the average detection time is about 70 ms, compatible with perception data rate (80 ms). Due to the camera perspective, VOD precision is proportional to object distance, with a measured range from ± 1 cm at 4 m to ± 2 m at 80 m.
- Rear View. One of the most important results to be considered for this application is the reliability level. Rear view application has reached a very low level (almost zero) of false positives when observer is slowly moving or not moving at all, i.e., the normal use of such application. However, false positives number slightly grows when the observer is moving at speeds higher than 20 km/h. The main issue related to false positives detection is the incoming vehicle lane-level localization. Even if the application provides such information, some errors may occur, thus generating false positives or negatives. Correct detection occurrences are really high. Evaluations made on real tests with normal environmental light condition showed a detection range higher than 95%. The minimum guaranteed detection distance is 35 m: this distance could not be enough for the observer to correctly execute the maneuver. Therefore, the amount of vehicles detected in time should be considered greater than 90%. Another

TABLE III
TRACK STATISTICS

Distance	13 km
Time	18 min
Average speed	43.33 km/h
Max speed	70 km/h
GNSS outage	4 min 22” (20.7%)

critical issue is the environmental light, since backlight conditions can reduce the correct detection percentage to level lower than 50%.

- Road markings. A set of real tests was performed on urban environments to validate this system. These tests demonstrate that vision algorithm was able to localize stop lines and pedestrian crossings up to 15m. Unfortunately this range is not enough to react properly and exploit the correct maneuver, then a sensor fusion was implemented in order to extend the detection range. Using localization information and a preloaded map, road markings positions were estimated up to 50 meters. Thus, a heuristic has been used to validate the GPS detections with vision systems in the short range and also to refine the relative orientation of the stopline.
- Roundabout. Lateral laser scanners have been used to manage roundabout entries and incoming vehicle detection. Their small working range was just enough for such application that need a vehicle detection range from 0 to 20m. Roundabout application has demonstrated good results but it was affected by blinking detection. To solve this problem, a delay is introduced in the vehicle reaction. Thus, any detection stops the autonomous vehicle that starts again only if no other moving obstacles are detected for 800 ms. Real tests showed good performances and conformable behaviors.
- Traffic Lights Detection. The method was already presented in [28], [33], together with the obtained results. The algorithm was modified in order to search for traffic lights installed on the side of the road, and was tested in different scenarios. The main problems were observed in case of sun light directly in the camera: the detection is extremely weak in these cases. False positives rate is extremely low mainly because the algorithm was activated only close to traffic lights.
- Lane Detection. The lane markings detection algorithm is based on the system already presented in [27]: a filter was added to provide more smooth results.

  Different cameras were used during the test, with good detection results; problems were encountered only in approaching curves with a radius smaller than 50 meters: in these cases the visible part of the lane markings was not enough for a robust detection, even using cameras with wider field of view.

As described in the previous sections, the World Perception Server gathers information from several applications, thus its performance partially depends on them; results related only to WPS behavior have been reported separately.

WPS may introduce errors in the tracking of large obstacles on the edge of field of view: buildings or guard rails on the road side could be detected as an object moving at the same speed of the observer. Another important aspect is the speed estimation of detected objects, that is computed using the displacement of their center of mass. As the evaluation of centers of mass by perception algorithms can be quite noisy, sudden variations on object speeds can occur, thus introducing errors in the obstacles trajectories estimation. The WPS is designed to filter out such sudden variations, but the use of features instead of center of mass can provide a better speed estimation.

Fig. 10. A frame of the video available on YouTube. On the top a recording of the outside of the vehicle and the map is shown. On the bottom, the internal of the vehicle, the vehicle speed and steering wheel angle values and the activated systems.

The test was recorded from different points of view, and the results are available on YouTube VisLab channel (see Fig. 10).[1]

V. Conclusion

In this article we presented the autonomous driving test which took place in Parma in July 2013. The test was performed on the ring road and on the streets of Parma downtown, in presence of other unpredictable road users. The vehicle has proven to be able to drive autonomously, without any human intervention, sharing roads with other vehicles, and manage roundabouts, intersections, priority roads, stops, tunnels, crosswalks, traffic lights, highways, and urban roads. Enriched maps (with the addition of the position of pedestrian crossings, traffic lights, and guard rails) were used to plan the route, and also for the activation of perception systems; no visual maps of the environment were used: both dynamic and static obstacles were detected in real time and distinguished. The use of very precise GNSS receiver and map, as done in this test, will be carefully considered in the future: if on one hand map enrichment seems to be needed in order to strengthen the vision algorithms, on the other hand a continuous precise position relative to the map may be no longer necessary when introducing vision-based self-localization. Topological information are indeed necessary for routing, just as it happens nowadays with satellite navigation systems. The use of a localization on maps, similar to the one used by many satellite navigation systems, strengthened by the information coming from perception algorithms (ego position with respect to lane markings, but also stopline and pedestrian crossing) may be sufficient. In our opinion, the complete removal of the maps is not the most profitable option for several reasons: first, maps are widely available, and secondly maps are necessary for high-level planning, i.e., choosing the path to reach a certain destination.

Although extremely successful, the PROUD test was not intended to be a demonstration that autonomous driving has been

[1]https://www.youtube.com/watch?v=PiUZ5NCXu-c
https://www.youtube.com/watch?v=O31CoFZbYRE
https://www.youtube.com/watch?v=dmD6kqBjnLM

solved: this experience should be seen as a demonstration of the possibilities and perspectives of the described technology and as another step in the direction of large-scale use of autonomous vehicles. One of the aspects that need to be further investigated and developed is the driving efficiency and speed: the test was carried out considering safety as a priority, and the most complicated maneuvers were carried out at a reduced speed. This behavior is not always consistent with human driving style.

Plus some perception problems still have to be solved: the problem of merging in large and multi-lane roundabouts, where vehicles travel at high speeds, has not been completely solved. Additional tests in different weather conditions must be conducted: no rain, fog, or snow were managed during the test. A final consideration is the fact that BRAiVE was developed to be a test bed for autonomous driving and not as a product. Therefore, even if safety was a major concern during its development and testing, no effort has been devoted in making the system robust to failures. In the case of a sensor or a PC failure, the system is able to detect and signal it to the driver, but it does not trigger any emergency maneuver. In other words, the current design does not take into account any redundancy or system robustness, not even to cyber attacks.

The test was a great success since the vehicle was able to handle all the situations without any human intervention and highlighted some new challenges that will drive our research in the next future, as detailed in this paper.

References

[1] C. Thorpe, T. Jochem, and D. Pomerleau, "The 1997 automated highway free agent demonstration," in *Proc. IEEE ITSC*, Nov. 1997, pp. 496–501.

[2] H.-S. Tan, R. Rajamani, and W.-B. Zhang, "Demonstration of an automated highway platoon system," in *Proc. Amer. Control Conf.*, Jun. 1998, vol. 3, pp. 1823–1827.

[3] E. van Nunen, R. Kwakkernaat, J. Ploeg, and B. Netten, "Cooperative competition for future mobility," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1018–1025, Sep. 2012.

[4] i GAME Project, "The Grand Cooperative Driving Challenge," 2014. [Online]. Available: http://www.gcdc.net

[5] Y. Sun, G. Xiong, W. Song, J. Gong, and H. Chen, "Test and evaluation of autonomous ground vehicles," *Adv. Mech. Eng.*, vol. 2014, 2014, Art. ID. 681 326.

[6] U. Ozguner, K. A. Redmill, and A. Broggi, "Team TerraMax and the DARPA grand challenge: A general overview," in *Proc. IEEE Intell. Veh. Symp.*, Parma, Italy, Jun. 2004, pp. 232–237.

[7] S. Thrun *et al.*, "Stanley: The robot that won the DARPA grand challenge," *J. Field Robot.*, vol. 23, no. 9, pp. 661–692, Sep. 2006. [Online]. Available: http://dx.doi.org/10.1002/rob.20147

[8] Y.-L. Chen *et al.*, "TerraMax: Team Oshkosh urban robot," *J. Field Robot.*, vol. 25, no. 10, pp. 841–860, Oct. 2008.

[9] J. Xin, C. Wang, Z. Zhang, and N. Zheng, "China future challenge: Beyond the intelligent vehicle," *IEEE Intell. Transp. Syst. Soc. Newslett.*, vol. 16, no. 2, pp. 8–10, Apr. 2014.

[10] T. Mei *et al.*, "Development of intelligent pioneer unmanned vehicle," in *Proc. IEEE IV Symp.*, Jun. 2012, pp. 938–943.

[11] P. Cerri *et al.*, "Computer vision at the Hyundai autonomous challenge," in *Proc. IEEE Intell. Transp. Syst.*, Washington, DC, USA, Oct. 2011, pp. 777–783.

[12] A. Broggi, M. Bertozzi, and A. Fascioli, "ARGO and the MilleMiglia in Automatico tour," *IEEE Intell. Syst. Appl.*, vol. 14, no. 1, pp. 55–64, Jan./Feb. 1999.

[13] F. Saust, J. Wille, B. Lichte, and M. Maurer, "Autonomous vehicle guidance on Braunschweig's inner ring road within the Stadtpilot project," in *Proc. IEEE IV Symp.*, Jun. 2011, pp. 169–174.

[14] T. Nothdurft *et al.*, "Stadtpilot: First fully autonomous test drives in urban traffic," in *Proc. IEEE 14th ITSC*, Oct. 2011, pp. 919–924.

[15] S. Thrun, "What we're driving at," The Official Google Blog, Oct. 2010. [Online]. Available: http://googleblog.blogspot.com/2010/10/what-were-driving-at.html

[16] U. Franke *et al.*, "Making Bertha see," in *Proc. IEEE ICCVW*, Dec. 2013, pp. 214–221.

[17] J. Ziegler *et al.*, "Making Bertha drive—An autonomous journey on a historic route," *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 2, pp. 8–20, Summer 2014.

[18] M. Bertozzi, L. Castangia, S. Cattani, A. Prioletti, and P. Versari, "360 detection and tracking algorithm of both pedestrian and vehicle using fisheye images," in *Proc. IEEE Intell. Veh. Symp.*, Seoul, Korea, Jun. 2015, pp. 132–137.

[19] M. Bertozzi *et al.*, "VIAC: An out of ordinary experiment," in *Proc. IEEE Intell. Veh. Symp.*, Baden Baden, Germany, Jun. 2011, pp. 175–180.

[20] A. Broggi *et al.*, "Extensive tests of autonomous driving technologies," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1403–1415, Sep. 2013.

[21] A. S. Huang, E. Olson, and D. C. Moore, "LCM: Lightweight communications and marshalling," in *Proc. IEEE/RSJ Int. Conf. IROS*, 2010, pp. 4057–4062.

[22] M. Bertozzi *et al.*, "GOLD: A framework for developing intelligent-vehicle vision applications," *IEEE Intell. Syst.*, vol. 23, no. 1, pp. 69–71, Jan./Feb. 2008.

[23] D. Dolgov and S. Thrun, "Autonomous driving in semi-structured environments: Mapping and planning," in *Proc. IEEE ICRA*, Kobe, Japan, 2009, pp. 3407–3414.

[24] A. Vatavu, S. Nedevschi, and F. Oniga, "Real-time environment representation based on occupancy grid temporal analysis using a dense stereo-vision system," in *Proc. IEEE Int. Conf. Intell. Comput. Commun. Process.*, 2010, pp. 203–210.

[25] A. Geiger, J. Ziegler, and C. Stiller, "Stereoscan: Dense 3D reconstruction in real-time," in *Proc. IEEE IV Symp.*, Jun. 2011, pp. 963–968.

[26] H. Hassannejad, P. Medici, E. Cardarelli, and P. Cerri, "Detection of moving objects in roundabouts based on a monocular system," *Expert Syst. Appl.*, vol. 42, no. 9, pp. 4167–4176, Jun. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417415000470

[27] M. Felisa and P. Zani, "Robust monocular lane detection in urban environments," in *Proc. IEEE Intell. Veh. Symp.*, San Diego, CA, USA, Jun. 2010, pp. 591–596.

[28] M. Diaz-Cabrera, P. Cerri, and J. Sanchez-Medina, "Suspended traffic lights detection and distance estimation using color features," in *Proc. IEEE Intell. Transp. Syst.*, Anchorage, AK, USA, Sep. 2012, pp. 1315–1320.

[29] OpenStreetMap Foundation, Openstreetmap. [Online]. Available: http://www.openstreetmap.org

[30] D. Ferguson, M. Likhachev, and A. T. Stentz, "A guide to heuristic-based path planning," in *Proc. ICAPS Int. Workshop Planning Uncertainty Auton. Syst.*, Jun. 2005, pp. 9–18.

[31] A. Broggi, S. Debattisti, M. Panciroli, and P. P. Porta, "Moving from analog to digital driving," in *Proc. IEEE Intell. Veh. Symp.*, Gold Coast, Qld., Australia, Jun. 2013, pp. 1113–1118.

[32] A. Broggi, A. Coati, P. Medici, M. Panciroli, and P. Zani, "Autonomous vehicles control in the VisLab intercontinental autonomous challenge," *Annu. Rev. Control*, vol. 36, no. 1, pp. 161–171, Apr. 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1367578812000132

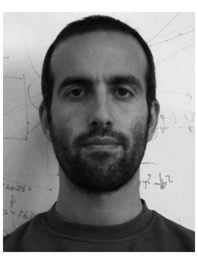
[33] M. Diaz-Cabrera and P. Cerri, "Traffic light recognition during the night based on fuzzy logic clustering," in *Computer Aided Systems Theory—EUROCAST*, vol. 8112, Lecture Notes in Computer Science, R. Moreno-Díaz, F. Pichler, and A. Quesada-Arencibia, Eds. Berlin, Germany: Springer-Verlag, Sep. 2013, pp. 93–100.

Alberto Broggi received the Dr.Ing. (master's) degree in electronic engineering and the Ph.D. degree in information technology both from the Università degli Studi di Parma, Parma, Italy, in 1990 and 1994, respectively. He is currently a Full Professor with the Università degli Studi di Parma, where he is also the President and CEO of the VisLab spinoff company. He is an author of more than 150 publications on international scientific journals, book chapters, and refereed conference proceedings. Dr. Broggi served as Editor-in-Chief of IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS for the term 2004–2008 and he served the IEEE Intelligent Transportation Systems Society as President for the term 2010–2011.

Pietro Cerri received the Dr.Eng. degree in computer engineering from the Università di Pavia, Pavia, Italy, in 2003 and the Ph.D. degree in information technologies from the Università degli Studi di Parma, Parma, Italy, in 2007. He was an Assistant Professor at the Artificial Vision and Intelligent System Laboratory (VisLab), Dipartimento di Ingegneria dell'Informazione, Università degli Studi di Parma. His researches are mainly focused on computer vision and sensor fusion approaches for the development of advanced driver assistance systems.

Stefano Debattisti received the master's degree in computer engineering in 2009 from the Università degli Studi di Parma, Parma, Italy, where he has been working toward the Ph.D. degree with the Dipartimento di Ingegneria dell'Informazione since 2010. His research is mainly focused on computer vision, path planning, and system architectures for advanced driver assistance systems.

Maria Chiara Laghi received the master's degree in computer engineering and the Ph.D. degree in information technologies from the Università degli Studi di Parma, Parma, Italy, in 2004 and 2010. She is currently a Research Engineer with VisLab, a spin-off company of the Università degli Studi di Parma. Her research activity focuses on intelligent systems for traffic analysis.



Paolo Medici was born in Reggio Emilia, Italy, in June 1978. He received the M.Sc. degree in electronic engineering and the Ph.D. degree in information technology from the Università degli Studi di Parma, Parma, Italy, in 2004 and 2009, respectively. Since 2004, he has been a Temporary Researcher with the Artificial Vision and Intelligent System Laboratory (VisLab), Dipartimento di Ingegneria dell'Informazione, Università degli Studi di Parma. His researches are focused on computer vision, calibration techniques, ensemble learning, and DSP programming, for the development of advanced driver assistance systems.

Daniele Molinari received the master's degree in computer engineering in 2013 from the Università degli Studi di Parma, Italy, where he is currently working toward the Ph.D. degree with VisLab. His main interests include computer vision, path planning, and vehicle dynamic and control.

Matteo Panciroli received the M.Sc. degree (*summa cum laude*) in information engineering and the Ph.D. degree in information technologies from the Università degli Studi di Parma, Parma, Italy, in July 2008 and 2012, respectively. Since 2010, he has been in charge of the sensor calibration task during the VIAC project. His research activity fields are autonomous vehicle and advanced driver assistance systems, and it is focused on sensor calibration and 3-D vision applications.

Antonio Prioletti received the master's degree in computer engineering in 2012 from the Università degli Studi di Parma, Parma, Italy, where he is currently working toward the Ph.D. degree with VisLab. He was a Visiting Scholar at the Computer Vision and Robotics Research Laboratory, University of California, San Diego, CA, USA. His main interests include computer vision, object classification, and data fusion for advanced driver assistance systems.