FREIE UNIVERSITÄT
BERLIN

Driving an autonomous car with eye tracking

Miao Wang, David Latotzky

B-10-03
March 2010

FACHBEREICH MATHEMATIK UND INFORMATIK
SERIE B • INFORMATIK

Driving an autonomous car with eye tracking

Miao Wang, David Latotzky
Free University of Berlin
Department of Computer Science
14195 Berlin, Germany
{mwang, latotzky}@mi.fu-berlin.de

Abstract

This paper describes eyeDriver, a hardware and software setup to drive an autonomous car with eye movement. The movement of the operator's iris is tracked with an infrared sensitive camera built onto a HED4 interface by SMI. The position of the iris is then propagated by eyeDriver to control the steering wheel of "Spirit of Berlin", a completely autonomous car developed by the Free University of Berlin which is capable of unmanned driving in urban areas.

1 Introduction and Motivation

Autonomous cars are robots capable of unmanned driving in an unknown urban area. Research in autonomous cars have received a broader interest in recent years and unfolded many insights for robot systems in different applications. The industry, especially automobile manufacturers, are eager to improve advanced driver assistance systems (ADAS), such as lane departure warning and intelligent speed adaptation systems while the military is strongly interested in unmanned drones for military purposes. From a Computer Science perspective autonomous vehicles function as a research foundation for progress in machine learning, computer vision, fusion of sensor data, path planning, decision-making and intelligent autonomous behavior.

Human driving can best be characterized with smooth maneuvers and driving with foresight. Autonomous driving up to today is still abrupt due to the underlying engines and discrete perceptions. Between human driving and unmanned driving there lies an area in which human behavior can be used as a guideline for autonomous control. EyeDriver is a direct implementation to use the eye movement of an operator to determine lateral steering commands to the car. Additionally, eyeDriver is a direct spin-off project to iDriver, a remote control for autonomous cars with an iPhone [1].

The eye tracking solution is based on helmet construction with two cameras and an infrared spotlight. The spotlight and one camera are directly aligned to illuminate and capture the operator's eyes. The captured grey images can be used to track the user's iris movement. The other camera is aligned to the operator's viewing direction providing a colored image of the user's field of view. After an initial calibration process a software module tracks the iris position over time and throws out outliners to provide a smooth tracking. The tracked positions are then mapped to steering commands for the steering wheel that are send out over UDP packets to the car.

The remainder of this paper is structured as follows. In Section 2, we give information on the background of autonomous cars in the DARPA Challenges and also describe the architecture of "Spirit of Berlin". Section 3 explains the user interface of the eye tracking software eyeDriver. Section 4 will describe the communication from the software to the car. Finally, Section 5 presents our conclusions and potential future use of this work.

2 Related Work

2.1 DARPA Grand Challenge

The US Defense Advanced Research Projects Agency (DARPA) has organized three challenges for unmanned land vehicles in 2004, 2005 and 2007.

In 2004, the goal was to drive a predefined route of 150 miles within 10 hours in Barstow, California. Price money of 1 million dollar was awarded for the team who finished first. Over 100 teams participated in this challenge, but none of them managed to complete the whole distance to win the price money. The best performance was accomplished by Carnegie Mellon Red Team's robot Sandstorm with 7.36 miles before crashing with a road obstacle [2].

DARPA repeated the same challenge in 2005, with 195 applicants whereas 43 where chosen for a National Qualification Event (NQE) and 23 teams made it to the finals. All but one of the finalists surpassed the distance of Sandstorm in the preceding year. Five vehicles successfully completed the race with the winner robot Stanley from Stanford Racing Team that finished the course in under 7 hours [3].

In 2007, DARPA moved to a urban scenario. The Urban Challenge in November 2007 took place at the site of the now-closed George Air Force Base in Victorville, California. The goal involved finishing a 60 mile course in urban area in less than 6 hours including obeying all traffic laws while negotiating with other traffic participants and obstacles and merging into traffic. The winner was Tartan Racing (Carnegie Mellon University and General Motors Corporation) with their vehicle Boss and a finishing time of 4 hours and 10 minutes [4].

While the 2004 and 2005 events were more physically challenging for the vehicles, because the robots only needed to operate in isolation with focus on structured situations such as highway driving, the 2007 challenge required engineers to build robots able obey all traffic regulations and make intelligent decisions in real time based on the current situation.

2.2 Spirit of Berlin

"Spirit of Berlin" was the participating robot of Team Berlin of the Free University of Berlin in the 2007 DARPA Urban Challenge [5]. It finished as one of the 35 semifinalists. The 2007 team was a joint team of researchers and students from Free University of Berlin, Rice University, and the Fraunhofer Society working together with American partners. The vehicle was a retrofitted Dodge Caravan with drive-by-wire technology, modified so that a handicapped person could drive using a linear lever for brake and gas (the lever controls all intermediate steps between full braking and full acceleration), and a small wheel for steering the front wheels. The rest of the car's components can be controlled through a small contact sensitive panel or using a computer connected to A/D converters.

Several sensors are used and mounted on top of the car: Two GPS antenna give information about the position and direction of the car. An IMU unit and an odometer provide temporal positioning information when GPS signal is lost. Two video cameras are mounted in front of the car for stereo-vision modules to detect lane markings and roadside. Three laser scanners are used to sweep the surroundings for any obstacles that need to be evaded. All sensor data are collected and fusioned into one state model about the vehicle itself and its surroundings that is representing the perception of the car.

A blade server from IBM provides the necessary computing power for the running software modules, the actual intelligence of the vehicle. Here, the fusioned sensor data are used to make decisions on what action needs to be executed next in the current situation. The necessary commands are then transmitted to actuators in the steering wheel, gas and brake pedals to execute the made decision. When eyeDriver is active, the intelligence in the car is turned off, and the steering commands given from the eye tracking are directly transmitted to the actuators.

3 User Interface

The eye tracking solution is based on a HED4 helmet construction sponsored by SMI GmbH with two cameras and a infrared spotlight that is depicted in Figure 1. One camera was a low resolution grey scale camera that was directly aligned to the operator's eye together with the spotlight. The camera provided high-constrast images of the user's pupil that could be used for iris tracking. The other camera was a middle resolution color camera headed forward to represent the operator's field of view. Both cameras needed to be calibrated prior to guarantee a good tracking quality. This was done with SMI's calibrating and tracking software iView X on a Laptop with Windows XP.

The calibration process required choosing 5 distinct points in the front camera image where the operator should fix the eye at. The fixed iris positions are saved and used to computer the calibration for both cameras. The iView software is then able to determine the eyes gazing position as x,y-coordinates in the front camera image. This x,y-coordinates are then used by a software module to transform them into steering commands for the autonomus car.

Figure 1: HED4 helmet for eye tracking

4 Communication to car

The communication between the software and car is based on simple Wi-Fi. The transmission from the software to the car requires values for the desired steering wheel position which is packed in a UDP packet and then send out to the car. The steering wheel position simply consists of one float value between -1.0 (maximum left) and 1.0 (maximum right), the default value for brake is set to 0. As a safety precaution to prevent rapid turns the steering wheel position is limited to -0.7 to 0.7.

The x,y-coordinates of the iView software are mapped to steering commands in two ways: One method scales the coordinates directly into steering commands from -0.7 to 0.7. This results in very accurate and exact steering corrections with every eye movement. This enables the operator to lock onto an object in the distance with his eyes in order to head towards it, making the persuit of a moving object very easy. However with this method, little shivering in the iris position is also interpreted as fluctuating corrections for the steering that needed to be smoothed out. Concentrating on a single object, especially if it is moving, also lessens the chance of distractions occuring, ensuring a more safe and smooth driving-experience.

The other method included using three steering windows, e.g. looking to the right means driving with a constant steering position to the right (left vice versa) and looking forward means driving straight. Unfortunately, it became apparent that this method was unintuitive for the operator as he could not fix any object in the distance for orientation. The simplification to relative orientation in three distinct areas proved to be more complicated to the operator than imagined. Since the 'borders' between the steering windows are invisible, keeping the eyes in the forward-window was harder than expected. Moving objects on the roadside easily distract the eyes if the driver tries to stare into a specific steering window without anything to focus on, making this method prone to fluctuations in the driving-direction. Especially when driving in a curve with many objects moving through the drivers field of view, maintaining a steady viewing direction was very exhausting. Therefore the first method was chosen with a simple smoothing algorithm to prevent fluctuating corrections.

5 Conclusion

This paper has described the architecture and mechanics of the eyeDriver software module to control a car's steering wheel with eye movement only. A HED4 helmet solution from SMI was used to track the iris movement of an operator after a calibration process. The iris position are mapped to x,y-coordinates of the front camera image and then transformed into steering wheel commands to the car. Two methods for transformation where presented: direct transformation and the usage of steering windows whereas the latter one proved to be unintuitive for an operator. The direct method allows the user to fix a target location in the distance to which the car will then drive to.

We see our work on eyeDriver as a proof-of-concept on which more applications can be built upon. While complete steering with the car is feasible but exhausting over some time, eye control can be limited to key areas where decisions need to be made (i.e. intersections with a choice between left and right).

References

[1] M. Wang and T. Ganjineh, "Remote Controlling an Autonomous Car with an iPhone," tech. rep., Free University of Berlin, 2009.

[2] C. Urmson, J. Anhalt, M. Clark, T. Galatali, J. P. Gonzalez, J. Gowdy, A. Gutierrez, S. Harbaugh, M. Johnson-Roberson, H. Kato, P. Koon, K. Peterson, B. Smith, S. Spiker, E. Tryzelaar, and W. Whittaker, "High speed navigation of unrehearsed terrain: Red team technology for grand challenge 2004," Tech. Rep. CMU-RI-TR-04-37, Carnegie Mellon University, June 2004.

[3] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney, "Stanley: The robot that won the darpa grand challenge," *Journal of Field Robotics*, vol. 23, pp. 661 – 692, September 2006.

[4] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. N. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer, M. Gittleman, S. Harbaugh, M. Hebert, T. M. Howard, S. Kolski, A. Kelly, M. Likhachev, M. McNaughton, N. Miller, K. Peterson, B. Pilnick, R. Rajkumar, P. Rybski, B. Salesky, Y.-W. Seo, S. Singh, J. Snider, A. Stentz, W. Whittaker, Z. Wolkowicki, J. Ziglar, H. Bae, T. Brown, D. Demitrish, B. Litkouhi, J. Nickolaou, V. Sadekar, W. Zhang, J. Struble, M. Taylor, M. Darms, and D. Ferguson, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics*, vol. 25, pp. 425 – 466, July 2008.

[5] J. Rojo, R. Rojas, K. Gunnarsson, M. Simon, F. Wiesel, F. Ruff, L. Wolter, F. Zilly, N. Santrac, T. Ganjineh, A. Sarkohi, F. Ulbrich, D. Latotzky, B. Jankovic, G. Hohl, T. Wisspeintner, S. May, K. Pervoelz, W. Nowak, F. Maurelli, and D. Droeschel, "Spirit of Berlin: An Autonomous Car for the DARPA Urban Challenge - Hardware and Software Architecture," tech. rep., Free University of Berlin, June 2007.