1997

Image Processing Based Control for Scaled Automated Vehicles

N. Schlegel
Qualcomm Incorporated

Pushkin Kachroo
University of Nevada, Las Vegas, pushkin@unlv.edu

Joseph A. Ball
University of Nevada, Las Vegas

J. S. Bay
Virginia Tech

IMAGE PROCESSING BASED CONTROL FOR SCALED AUTOMATED VEHICLES

Nikolai Schlegel
Qualcomm, Inc.
6455 Lusk Blvd. , BB-102H
San Diego, CA 92121

Pushkin Kachroo
Joseph A. Ball
John S. Bay
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061

Keywords: Autonomous vehicle, H-Infinity Control, Image Processing

ABSTRACT

This paper presents a way to design a lateral controller for an automated vehicle using information gained through image processing with the control objective being to stay on a desired path. Two possible ways to obtain the information necessary for lateral control by image processing are presented, one based on pixel intensity summation and the other on vanishing point calculations. The paper also describes two algorithms for the actual lateral control, one based on classical control theory and the other on modern H_∞ control. The resulting control algorithms were implemented on a scaled autonomous vehicle system.

1. INTRODUCTION

The scaled automated vehicle described in this paper is part of the FLASH[1]-Laboratory at the Virginia Polytechnic Institute and State University (see [1]), a research lab equipped with a scaled model highway and a number of scaled model vehicles (approximate scale 1:15).

In the following sections, the hardware and software of the scaled vehicle will be briefly outlined, followed by an explanation of the two different image processing algorithms used to determine the vehicle position. The first method is based on the the intensity of the pixels in a certain area of the captured image (intensity profile). The second method calculates the orientation of the vehicle with respect to the road by approximating the white middle marker with a straight line and determining where this line crosses the virtual horizon in the image plane (vanishing point). Finally, two different algorithms to do the actual lateral control are discussed. The first of these algorithms is based on a simply P-controller. The second method is based on finding an appropriate mathematical model for the vehicle and designing an H_∞ controller for it.

[1] Flexible-Low-cost-Automated-Scaled-Highway

2. THE VEHICLE

The model vehicle can either be operated by telerobotic commands from a human driver or, when equipped with a video camera, it can run in a fully autonomous mode. A complete description is given in [2].

2.1. Hardware

The basis for the small scale model vehicle is a toy remote control car (see Figure 1). This vehicle is be equipped with a video camera that records an image of the road ahead. Other modifications include the addition of a microcontroller (a

0-7803-4269-0/97/$10.00 © 1998 IEEE

68HC11) and a radio modem to communicate with a stationary computer over a wireless serial link. Most of the calculations are done on this stationary unit, including the image processing and the lateral control algorithm itself.

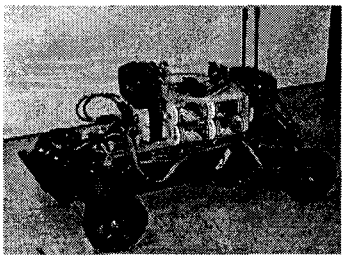

Figure 1: The automated model vehicle

2.2. Software

The task of the microcontroller is just to convert high-level commands from the stationary computer into low-level PWM signals to drive the actuators of the vehicle. This is done in a small C program that runs from the microcontroller memory.

The job of the C program on the stationary computer is to handle the image processing, the lateral control algorithm as well as to communicate with the mobile vehicle and possibly a human operator. For efficiency, the program is divided into separate tasks that run concurrently on a real-time multitasking kernel. The algorithms for image processing and lateral control are discussed in the following sections.

3. IMAGE PROCESSING

The goal for the image processing is to extract information about the position of the vehicle from a captured image of the road ahead of the vehicle. In both of the algorithms presented below, a white middle marker on an otherwise dark road is used as a reference for the vehicle to follow. Once the white middle marker is identified it can be used to derive information about the vehicle's location and orientation with respect to the road.

3.1. Intensity Profile

All this straight-forward approach does is sum up all pixel values in a certain area in the lower part of the captured image in the vertical direction, therefore getting a one-dimensional horizontal pixel intensity-profile of the image.

As seen in Figure 2, the white middle marker results in a peak in the computed horizontal intensity profile. All that has to be done now is find the peak intensity (with the x-coordinate x_{max}) and measure its distance from the center of the image (denoted by x_{middle}). This distance can then be used as an error signal $e = x_{max} - x_{middle}$ for a lateral controller that will produce the commands to move the steering angle of the front wheels. This algorithm used for detecting the middle marker has a good performance and is very fast, therefore potentially allowing high speeds of the vehicle.

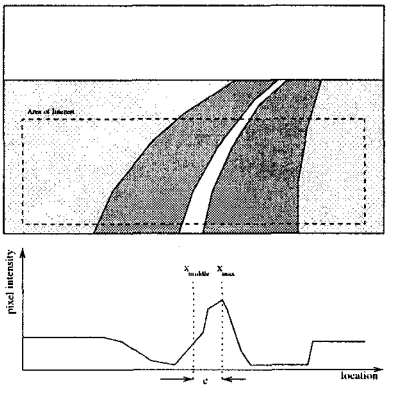

Figure 2: Pixel intensity profile

Although the above algorithm has actually been tested successfully with the vehicle, it has a number of disadvantages. First of all, is very sensitive to misinterpretations since it looks for peaks in the profile. Another disadvantage is that the distance that is returned as an error is not very precise since it is a combination of the lateral deviation of the vehicle's center to the white middle marker and the angle that the vehicle has to the marker. This might result in poor performance at curves..

3.2. Vanishing Points

The basic concept of vanishing point analysis is to obtain the orientation of the vehicle based on the location of some vanishing points in the image plane (see [3] for a full description). The orientation is defined by the heading- or yaw-angle φ_h and the lateral distance or deviation Δy.

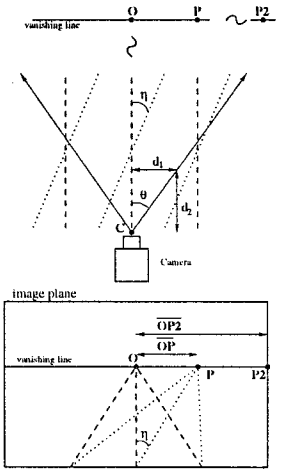

Figure 3: Lines & vanishing points

A vanishing point is the virtual intersection of a number of straight parallel lines at a great distance under perspective projection. The vanishing point O in Figure 3 is formed by all straight lines that are parallel to the principal axis of view (camera axis). The vanishing point P is formed by another set of parallel lines. From Figure 3 it is intuitively clear that there is a correspondence between the angle η and the distance $\overline{OP}$ between O and P. Assuming that both the distance $\overline{OP_2}$ on the vanishing line and the angle θ are known, the angle η can be expressed as a function of the distance $\overline{OP}$.

$$\eta = \tan^{-1}\left(\frac{\overline{OP} * \tan\theta}{\overline{OP_2}}\right) \qquad (1)$$

Determining the vanishing point p is done by approximating the white middle marker in the captured image with a straight line extended to the horizon (using an edge detection algorithm that detects a transition from dark to bright and back to dark). The heading angle φ_h is then identical to the angle η in equation (1). A similar equation can be used to calculate the lateral deviation Δy (again, see[3] for details).

4. LATERAL CONTROL

4.1. P-Controller

The whole idea of a P-controller is to multiply a control error e with a constant gain. In case of the intensity profile algorithm as the image processing method, the control error is the distance $x_{max} - x_{middle}$ between the peak of the intensity profile and the middle of the image.

In the case of the vanishing point analysis there are two control errors, φ_h and Δy. Since the P-controller is like all classical controllers a SISO system, the two control errors of the vanishing point analysis have to be combined to one value. This can be done by making e a linear combination of φ_h and Δy. This combining however means a loss of information. On the other hand, testing the P-control algorithm in autonomous driving mode resulted in a surprisingly good performance, considering it simplicity.

4.2. H_∞-Controller

One big advantage of H_∞ control is that it guarantees stability of the plant it controls, even if the plant contains some uncertainties like unknown parameters. The only assumption that has to be made is that these uncertainties have to be bounded, or more precise, the worst-case deviation of the plant due to the uncertainties has to be known. This property of H_∞ control is also called robust stabilization.

The first step in designing a controller is the derivation of a plant model. Here, the dynamics of the vehicle were modeled using a classical linearized single-track model (see [4] for a detailed description). The model is a four state, one input (steering angle δ_f),

two output (heading angle φ_h and lateral deviation Δy) system normally represented in state-space form.

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \\ x &= [\ \beta \quad r \quad \varphi_h \quad \Delta y\ ]', u = [\delta_f] \end{aligned} \tag{2}$$

Apart from φ_h and Δy, which are both states and outputs, the systems also has the side-slip angle β and the yaw rate r as states. Note that due to the integrative nature of the plant (steering angle determines the rate of direction change), it has a double pole at zero.

The plant is exposed to several disturbances that make the job for a control algorithm more difficult (Figure 4).

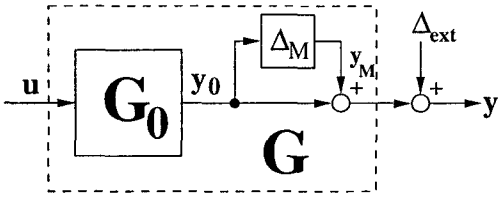

Figure 4: Disturbances acting on the plant

The model for the vehicle dynamics introduced earlier always assumes that the road is straight. Curves in the road can therefore be modeled as external disturbances Δ_{ext} to the plant. One design goal is therefore the design for performance: The external disturbances should have a frequency dependent effect on the output of the plant. For low frequencies, the effect of the disturbances should be maximized (the vehicle should follow the curve) whereas for high frequencies, the effect of the disturbances should be minimized (high frequency disturbances are most likely measurement errors). This design goal can be expressed as a frequency dependent weighting of the sensitivity function S with $S = (I + GK)^{-1}$ (see [5]).

Even though there is a mathematical model for the plant, the parameters of this model might not be fully known. This can be modeled as a multiplicative uncertainty. The real plant G then can be modeled as $G = G_0(I + \Delta_M)$. Since this multiplicative uncertainty also takes the form of a disturbance, it is also necessary to design for robust stability: The uncertainties in the plant due to parameter variations should have a minimal effect on the control signal u (the output of the controller). In other words, the complementary sensitivity function T with $T = GK(I + GK)^{-1} = 1 - S$ should be minimized (see [5]).

For the H_∞ controller design, the mixed sensitivity approach is used [5, 6]. In H_∞ terms, the goal of the mixed sensitivity approach can be specified as follows:

$$\left\| \begin{matrix} W_1 S \\ W_3 T \end{matrix} \right\|_\infty < \gamma \tag{3}$$

The H_∞ norm of $W_1 S$ and $W_3 T$ has to be smaller than some upper bound γ at all times and at all frequencies. Here W_1 and W_3 are some frequency dependent weighting functions that can be used to further specify a design goal for the two transfer functions.

A standard way of designing a H_∞-controller is by solving the generalized regulator problem [6]. In this problem, there exists a generalized plant P with four signals w, u, z and y. The signal w contains all exogenous inputs and model-error outputs, the signal u is the controller output (and therefore a plant input) and the signal y is the controller input (and therefore a plant output). The signal z of dimension is called the objective signal.

The goal for the generalized regulator problem is to find a controller K for the generalized plant P that will minimize the transfer function R_{zw} that maps the input w to the objective z. An additional constraint is that the closed loop of P and K has to be stable. In H_∞ terms, this can be ex-

pressed as $||R_{zw}||_\infty < \gamma$ and K stabilizes P In this case, the input signal w consists of a 2-dimensional exogenous input that lumps together the effect of the external disturbance Δ_{ext} and the effect of the multiplicative uncertainty Δ_M on the 2-dimensional plant output y consisting of the heading angle φ_h and the lateral deviation Δy. The control signal u is the 1-dimensional steering angle δ_f. The objective signal z is the signal of equation 3 (consisting of $W_1 T$ and $W_3 S$) that has to be bounded.

For the matrices of the generalized plant, a number of standard assumptions are made [6]. These have to be met for a stabilizing controller of the generalized plant to exist. One of the these assumptions is that P has full rank for all ω.

The choice of the weighting functions W_1 and W_3 has to reflect the design goals specified earlier. The design goal for the sensitivity function S was that low-frequency disturbances have to pass through to the output without any change while high-frequency disturbances should be damped.

When choosing the bound for the H_∞ norm to be $\gamma = 1$, this weighting function ensures that

$$||S||_\infty \leq \left|\left|W_1^{-1}\right|\right|_\infty \qquad (4)$$

therefore enforcing the specified design goal of damping high-frequency disturbances.

The weighting function W_3 can be found by looking at the variation of the plant output when picking different values for the plant parameters. Due to the poles of the plant at the origin (see above), P does not have full rank for $\omega \to \infty$. This problem can be solved by introducing a double differentiator in W_3, making P full rank at ∞. Such a weighting function enforces the design goal of robust stabilization for the transfer function T that maps $y \to u$.

$$||T||_\infty \leq \left|\left|W_3^{-1}\right|\right|_\infty \qquad (5)$$

Removing the poles at the origin for the purpose of deriving the controller can be done using a bilinear transform, as shown in [7, 8] for the stabilization of an double integrator. By using the bilinear transform

$$s = \frac{\tilde{s} + p_1}{\frac{\tilde{s}}{p_2} + 1} \qquad p_1 < 0 \quad p_2 = \infty \qquad (6)$$

in the frequency domain, the imaginary axis in the s-domain gets transformed into a circle with infinite radius located p_1 units to the right of the imaginary axis in the $\tilde{s}$-domain. This is equivalent to just shifting the imaginary axis by p_1 units to the left.

Obtaining a controller that solves the generalized regulator problem normally requires solving of two algebraic Riccati equations. One solutions forms a full state feedback controller while the second solution is used to create an observer that estimates the states of the plant from the actual output.

In [6] such a controller is designed based on the solution of two Riccati equations. However, the authors also mention several special cases in which solving only one Ricatti equation is sufficient. In our case the output of the generalized plant can be written as $y = Cx + w$ (no u term!). This means that an observer can be created that perfectly reconstructs all the states x and exogenous inputs w from the output y, therefore it is only necessary to solve the first algebraic Ricatti equation.

A H_∞ controller K can then be synthesized for the plant meeting all the previously specified design goals. Note that the controller is actually designed for the transformed plant (see equation 6). However, by applying the inverse transformation on the resulting H_∞ controller, a controller for the original plant can be found.

The simulation in Figure 5 shows that the controller is capable of not only stabilizing

a nominal plant but also one with a multiplicative uncertainty (reflecting uncertainties in the plant model). For a comparison, Figure 6 shows the plant without a controller. The plot clearly shows that the plant is unstable, corresponding to real-world experience that an uncontrolled vehicle will eventually run off the road.

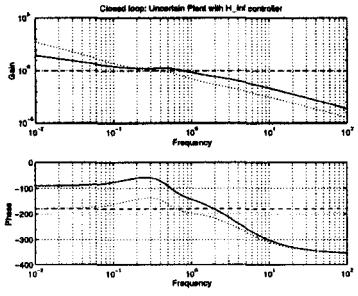

Figure 5: Closed loop with uncertainties

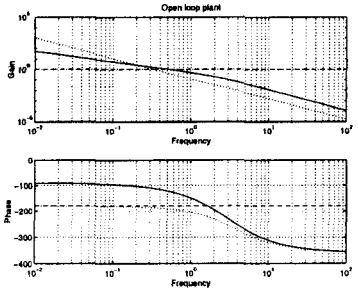

Figure 6: Open loop response

Unfortunately the encouraging results of the simulation didn't translate to good performance in real live. When the algorithm was implemented on the small-scale vehicle, the autonomous steering tends overcontrol on many occasions (especially in curves), causing the vehicle to go of the road.

5. CONCLUSIONS

This paper presented a small-scale autonomous vehicle system for the FLASH-Lab model vehicle park capable of being operated in a fully automated mode with the aid of image processing. For the image processing part, two algorithms have been tested. Both of them work well and provide information about the position and orientation of the vehicle with respect to the road. The first lateral control algorithm, relying on a simple classical control law, is working reliably and can be used for automated driving at moderate speed. The second control algorithm, based on a model with uncertain parameters in combination with H_∞ control, doesn't show the same performance and reliability as the other, more simple design. While definitely having the greater potential due to its robust stabilization and greater complexity, at this stage the H_∞ control algorithm lacks the necessary performance for automated driving mode and requires future fine-tuning.

References

[1] Pushkin Kachroo, *Setup for advanced vehicle control systems experiments in the flexible low-cost automated scaled highway (FLASH) laboratory*, SPIE's Photonics East Symposium, Mobile Robots X, Philadelphia, PA, 1995

[2] Nikolai Schlegel, Pushkin Kachroo, *Telerobotic operation combined with Automated Vehicle Control*, SPIE's Photonics East Symposium, Mobile Robots X, Boston, MA, 1996

[3] E.C. Yeh, J.-C. Hsu, R.H.Lin, *Image-Based Dynamic Measurement for Vehicle Steering Control*, Intelligent Vehicles Symposium Proceedings 1994, P.326-332, IEEE, Piscataway, NJ,1994

[4] Pushkin Kachroo, *Nonlinear control strategies and vehicle traction control*, UMI, Ann Arbor, MI, 1993

[5] J. Ball, . Rakowski, *Interpolation by Rational Matrix Functions and Stability of Feedback Systems: the 2-Block-Case*, Journal of Mathematical Systems, Estimation and Control, Vol 4, Birkh'auser, Boston, 1994

[6] M. Green, D.J.N. Limebeer, *Linear Robust Control*, Prentice Hall, Eaglewood Cliffs, NJ, 1995

[7] Richard Y. Ching, Michael G. Safonov, *Robust Control Toolbox*, The MathWorks Inc., 1992

[8] Michael G. Safonov, *Imagrinary-Axis in Multivariable H_∞-Optimal Control*, NATO ASI Series, Vol. F34, Springer Verlag, Heidelberg, 1987