

RALPH: Rapidly Adapting Lateral Position Handler

Dean Pomerleau

pomerlea@ri.cmu.edu

Phone: 412-268-3260 Fax: 412-268-5571

The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA

and

AssistWare Technology, 2498 Elkridge Drive, Wexford, PA 15090, USA

Introduction

Nearly 15,000 people die each year in the US in single vehicle roadway departure crashes[11]. These accidents are often caused by driver inattention, or driver impairment (e.g. fatigued or intoxicated drivers). A system capable of warning the driver when the vehicle starts to depart the roadway, or controlling the lateral position of the vehicle to keep it in its lane, could potentially eliminate many of these crashes. Nearly 70% of these crashes occur in rural or suburban settings on undivided two lane roads[11]. Since it is unlikely these roads will be upgraded in the foreseeable future, a system for preventing these crashes must rely on the existing road structure.

Research into such systems has focused on machine vision techniques that detect particular features in video images of the road ahead of the vehicle, and determine the desired vehicle trajectory based on the relative positions of these features. Many of these systems [2][4][5][7] rely on tracking specific features, such as lane markings, from one image to the next. Others depend on detecting regions of the image representing the road based on features such as color [1][6] or texture [12].

All these systems have a common characteristic. They all have a strong, a priori model of the road's appearance, and employ hand programmed detection algorithms to locate these characteristic features.

Unfortunately, roads are not always cooperative. Road markings vary dramatically depending on the type of road (e.g. suburban street vs. interstate highway), and the state or country in which it is located. For example, many California freeways use regularly spaced reflectors embedded in the roadway, not painted markings, to delineate lane boundaries. Further challenges result from the fact that the environmental context can greatly impact road appearance. Changes in illumination due to shadows, glare or darkness, and obstructions by other vehicles, rain, snow, salt or other foreign objects often cause dra-

matic changes in the road's appearance. Together these variations often invalidate the assumptions underlying vision algorithms, resulting in poor road detection performance.

Alternative approaches that combine machine vision and machine learning techniques have demonstrated an enhanced ability to cope with variations in road appearance [4][8][9]. ALVINN is a typical system of this type. ALVINN employs an artificial neural network to learn the characteristic features of particular roads under specific conditions. It utilizes this learned road model to determine how the vehicle should be steered in order to remain in its lane. While systems of this type have been quite successful at driving on a wide variety of road types under many different conditions, they have several shortcomings. First, the process of adapting to a new road requires a relatively extended "retraining" period, lasting at least several minutes. While this adaptation process is relatively quick by machine learning standards, it is unquestionably too long in a domain like autonomous driving, where the vehicle may be travelling at nearly 30 meters per second. Second, the retraining process invariably requires human intervention in one form or another. These systems employ a supervised learning technique such as backpropagation, requiring the driver to physically demonstrate the correct steering behavior for the system to learn.

A truly flexible system should 1) flexibly exploit whatever features are available to determine vehicle location, 2) adapt almost instantly when the available features change, and 3) perform this adaptation without human supervision. This paper presents a system called RALPH (Rapidly Adapting Lateral Position Handler) which demonstrates these characteristics.

RALPH Processing

RALPH decomposes the problem of steering a vehicle into three steps, 1) sampling of the image, 2) determining the road curvature, and 3) determining

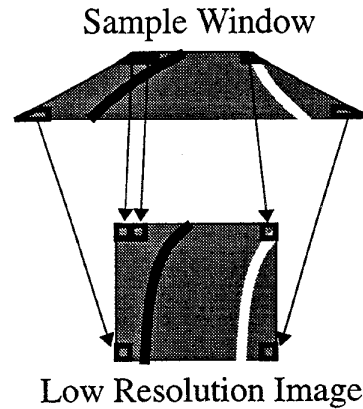
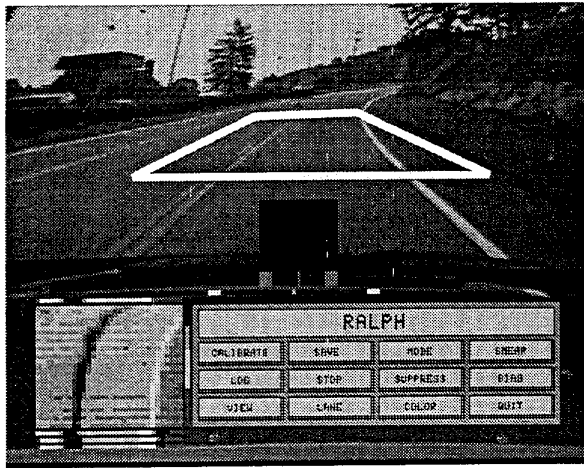


Figure 1: Image from the forward looking camera (left), and RALPH's image sampling strategy (right).

the lateral offset of the vehicle relative to the lane center. The output of the later two steps are combined into a steering command, which can be compared with the human driver's current steering direction as part of a road departure warning system, or sent directly to the steering motor on our Navlab 5 testbed vehicle for autonomous steering control[3].

Image Sampling

A typical scene of the road ahead, as imaged by a video camera mounted next to the rearview mirror on our testbed vehicle, is depicted on the left of Figure 1. The scene depicted is a greyscale image, although RALPH can utilize either black and white or color images, using a color-based contrast enhancement technique described in [8]. Obviously much of this image is irrelevant for the driving task (e.g. the parts of the image depicting the sky or the dashboard of the vehicle). These parts of the scene are eliminated, and only the portions of the scene inside the white trapezoid are processed. While the lower and upper boundaries of this trapezoid vary with vehicle velocity (moving further ahead of the vehicle, towards the top of the image, as vehicle speed increases), they typically project to approximately 20m and 70m ahead of the vehicle, respectively.

The second, and perhaps more important aspect of the trapezoid's shape is its horizontal extent. It is configured so that its width on the groundplane is identical at each row of the image. The horizontal distance that each row of the trapezoid encompasses is approximately 7.0 meters, about twice the width of a typical lane. This trapezoid is selectively sampled according to the strategy depicted in the schematic on the right

of Figure 1 so as to create a low resolution (30x32 pixel) image in which important features such as lane markings, which converged towards to top of the original image, now appear parallel in the low resolution image. Note that this image resampling is a simple geometric transformation, and requires no explicit feature detection.

Determining Curvature

The "parallelization" of road features described above is crucial for the second step of RALPH processing, curvature determination. To determine the curvature of the road ahead, RALPH utilizes an "hypothesize and test" strategy. RALPH hypothesizes a possible curvature for the road ahead, subtracts this curvature from the parallelized low resolution image, and tests to see how well the hypothesized curvature has "straightened" the image. This process is depicted in Figure 2. In this example, five curvatures are hypothesized for the original image, shown at the top. For each of the five hypothesized curvatures, the rows of the image are differentially shifted in an attempt to "undo" the curve and straighten out the image features. For left curve hypotheses, rows are shifted towards the right and for right curve hypotheses, rows are shifted towards the left. For the more extreme hypothesized curvatures (on the far left and right), the rows of the original image are shifted further than for the less extreme curvatures (in the middle). For all the hypothesized curvatures, rows near the top of the image, corresponding to regions on the groundplane further ahead of the vehicle, are shifted further horizontally than rows near the bottom of the image. This differential shifting accounts for the fact that for a given hypothesized curvature, the road will be dis-

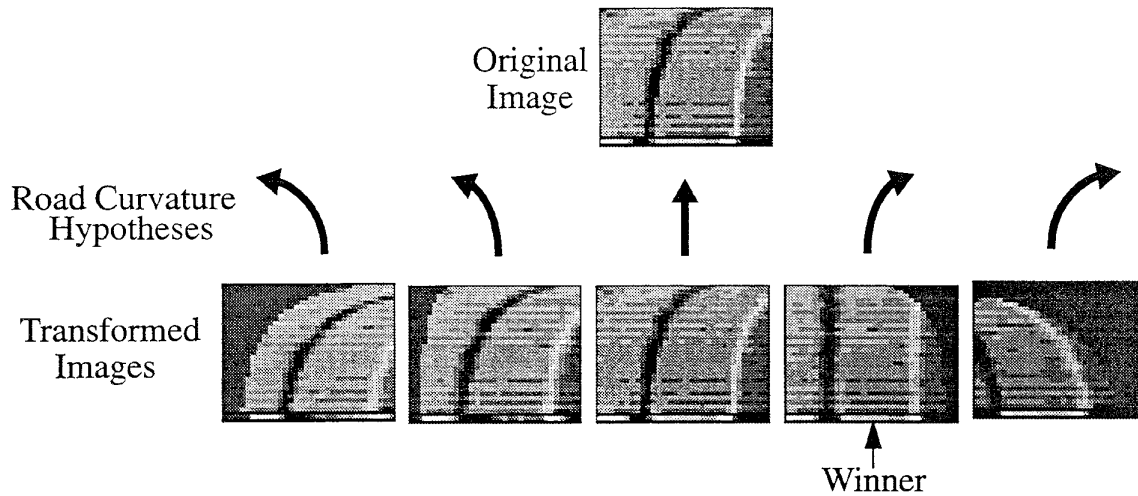


Figure 2: Technique for determining road curvature by “straightening” image features.

placed more at the top of the image, far ahead of the vehicle, then at the bottom. The exact shift distance for each row in the transformed images is determined both by the geometry of the camera, and the particular curvature hypothesis being tested.

As can be seen from Figure 2, the second curvature hypothesis from the right, corresponding to a shallow right turn, has resulted in a transformed image with the straightest features, and therefore should be considered the winning hypothesis. The technique used to score the “straightness” of each hypothesis is depicted in Figure 3. After differentially shifting the rows of the image according to a particular hypothesis, columns of the resulting transformed image are summed vertically to create a scanline intensity profile, shown in the two curves at the bottom of Figure 3. When the visible image features have been straightened correctly, there will be sharp discontinuities between adjacent columns in the image, as show in the right scanline intensity profile in Figure 3. In contrast, when the hypothesized curvature has shifted the image features too much or too little, there will be smooth transitions between adjacent columns of scanline intensity profile, as depicted in the left scanline intensity profile of Figure 3. By summing the maximum absolute differences between intensities of adjacent columns in the scanline intensity profile, this property can be quantified to determine the curvature hypothesis that best straightens the image features.

An important attribute to note about this technique for determining road curvature is that it is entirely independent of the particular features present in the image.

As long as there are visible features running parallel to the road, this technique will exploit them to determine road curvature. These features need not be located at any particular position relative to the road, and need not have distinct boundaries - characteristics required by systems that utilize strong a priori road models and edge detection.

Determining Lateral Offset

The next step in RALPH’s processing is to determine the vehicle’s lateral position relative to the lane center. This is accomplished using a template matching approach on the scanline intensity profile generated in the curvature estimation step. The scanline intensity profile is a one dimensional representation of the road’s appearance as seen from the vehicle’s current lateral position. By comparing this current appearance with the appearance of a template created when the vehicle was centered in the lane, the vehicle’s current lateral offset can be estimated.

Figure 4 illustrates this lateral offset estimation procedure in more detail. Here, the current scanline intensity profile is depicted on the left, and the template scanline intensity profile, generated when the vehicle was centered in the lane, is depicted on the right. By iteratively shifting the current scanline intensity profile to the left and right, the system can determine the shift required to maximize the match between the two profiles (as measured by the correlation between the two curves). The shift distance required to achieve the best match is proportional to the vehicle’s current lateral offset.

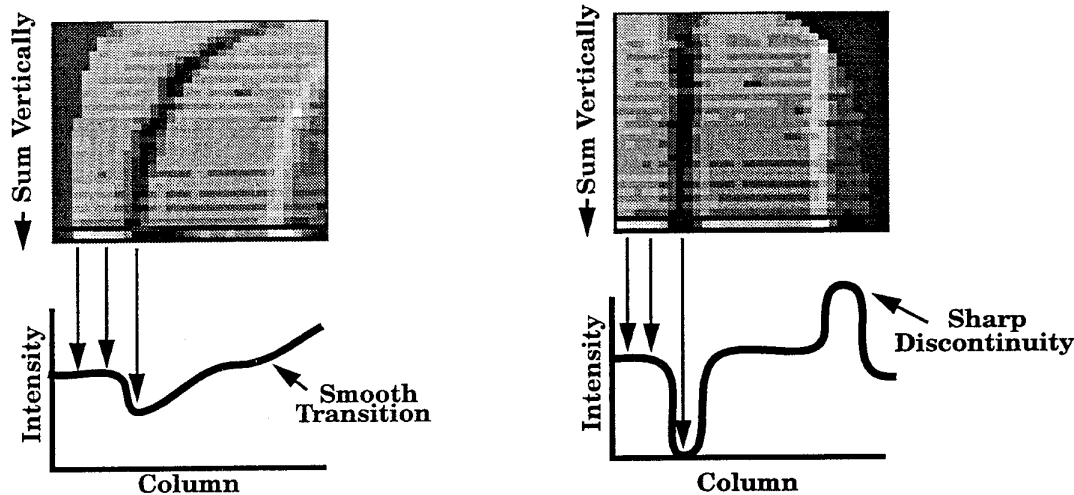


Figure 3: Technique for scoring curvature hypotheses based on straightness of resulting image features.

Note that as with the curvature determination step, this process does not require any particular features be present in the image. As long as the visible features produce a distinct scanline intensity profile, the correlation based matching procedure be able to determine the vehicle's lateral offset. In particular, even features without distinct edges, such as pavement discoloration due to tire wear or oil spots, generate identifiable scanline intensity profile variations which RALPH easily exploits to determine lateral offset. This is a performance feature which edge-based road detection systems do not share.

Adapting to Changing Conditions

Another important feature of RALPH stems from the simplicity of its scanline intensity profile representation of road appearance. The 32 element template scanline intensity profile vector is all that needs to be modified to allow RALPH to drive on a new road type. Modifying this vector is extremely easy. In the current RALPH implementation there are four ways of adapting the template to changing conditions.

The first method involves a human driver centering the vehicle in its lane, and pressing a button to indicate that RALPH should create a new template. In under 100 msec, RALPH performs the processing steps described above to create a scanline intensity profile for the current road, and then saves it as the default template. From that point on, RALPH can drive (or warn the driver of road departure danger) on this road using the newly created template to determine the vehicle's position relative to the lane center.

A second method for acquiring a template appropriate for the current road type is to select one from a library of stored templates recorded previously on a variety of roads. RALPH can select the best template for the current conditions by testing several of these previously recorded templates to determine which has the highest correlation with the scanline intensity profile created for the current image.

The third method of template modification occurs after an appropriate template has been selected. During operation, RALPH slowly "evolves" the current template by adding a small percentage of the current scanline intensity profile to the template. This allows the current template to adapt to gradual changes in the road's appearance, such as those caused by changes in the sun's angle.

RALPH handles more abrupt scene changes, such as changes in lane marker configuration, using the final and most interesting template modification strategy. In this technique, RALPH uses the appearance of the road in the foreground to determine the vehicle's current lateral offset and the curvature of the road ahead, as described above. At the same time, RALPH is constantly creating a new "rapidly adapting template" based on the appearance of the road far ahead of the vehicle (typically 70-100 meters ahead). This rapidly adapting template is created by processing the distant rows of the image in the same manner as described above. The roads curvature is assumed to be nearly constant between the foreground and background, allowing RALPH to determine where the road is ahead and therefore what the new template should look like when the vehicle is centered in its lane.

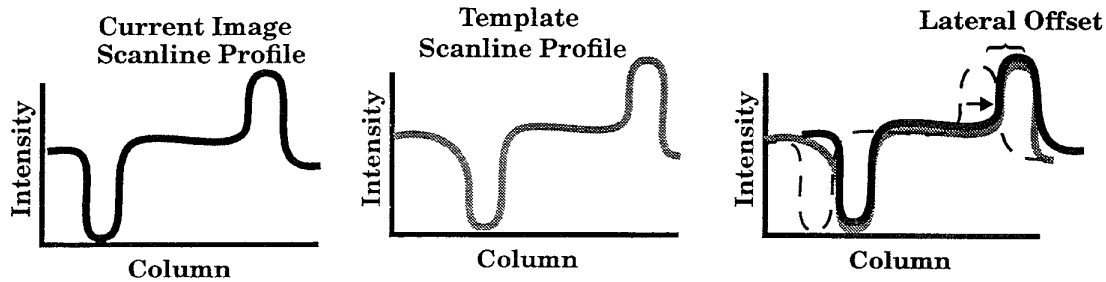


Figure 4: Technique for determining lateral offset by shifting scanline profile to match template.

If the appearance of the road ahead changes dramatically, RALPH uses this technique to quickly create a template appropriate for the new road appearance. When the vehicle actually reaches the new road, RALPH determines that the template it was previously using is no longer appropriate, since it does not match the scanline intensity profile of the current image. It therefore swaps in the rapidly adapting template, and continues driving. Note that this rapid adaptation occurs in the time span of approximately 2 seconds, without any human intervention.

Ralph Performance

Using the above techniques, RALPH has been able to locate the road and steer autonomously on a wide variety of road types under many different conditions. RALPH has driven our Navlab 5 testbed vehicle over 3000 miles on roads ranging from single lane bike paths, to rural highways, to interstate freeways. Part of this 3000 miles was a trip from Pittsburgh, PA to Washington, DC. During this 302 mile trip, RALPH steered autonomously 96% of the way. Times when intervention was required included preventing RALPH from following offramps, and changing lanes to pass slow moving vehicles. Velocity was controlled by the safety driver, and the vehicle's average speed during the trip was 57 mph. In another set of experiments, RALPH has driven at speeds of up to 91 mph on the TRC test track in East Liberty, Ohio.

RALPH has driven successfully in conditions including bright sun with harsh shadows, dense fog, rain, and nighttime using only headlight illumination. On numerous occasions, RALPH has demonstrated a flexibility not possible with previous lateral position estimation and control systems. For instance, several times glare off wet pavement has been severe enough to entirely obscure the lane markings in the video image. On those occasions RALPH has successfully exploited the tracks left on the pavement by previous vehicles to determine its lateral position and the road

curvature, allowing it to continue driving. As another example, when lane markers are worn or degraded, RALPH has demonstrated the ability to utilize the diffuse discoloration down the center of the lane, caused by oil spots from previous vehicles, to locate the road ahead and steer the vehicle.

RALPH has also demonstrated its ability to quickly adapt to dramatic changes in road appearance. Using the technique for rapidly adapting its template, RALPH can handle changes in the number of lanes on the road, as well as changes in the lane marker configuration. With this technique, RALPH has also driven through tunnels, which are perhaps the most difficult situation for vision-based road followers because of the accompanying large changes in lighting conditions and lane markings.

Future Work

The current focus on research on RALPH involves both extending its capabilities and quantifying its performance. For extensions, we are currently developing techniques which allow RALPH to change lanes automatically, as well as focus of attention methods which allow RALPH to ignore or follow offramps as appropriate.

In the area of performance quantification, we will shortly embark on a cross country trip, from Pittsburgh to Los Angeles, during which RALPH will steer autonomously as much as possible. We plan to record data on both on the percentage of time RALPH is able to steer correctly, and on the conditions in which manual intervention is required. As with the trip from Pittsburgh to Washington, we expect the percentage of autonomous travel to be over 95%, demonstrating RALPH's potential as a reliable system for autonomous control and roadway departure warning.

References

- [1] Crisman, J. and Thorpe, C. (1990) "Color vision for road following". *Vision and Navigation: The CMU Navlab*. C. Thorpe (ed.) Kluwer Academic Publishing, Boston MA.
- [2] Dickmanns, E. D., Behringer, R., Dickmanns D., Hildebrandt, T., Maurer, M., Thomanek, F., and Schielen, J., "The seeing passenger car 'VaMoRs-P,'" *1994 IEEE Symposium on Intelligent Vehicles*, pp. 68-73.
- [3] Jochem, T., Pomerleau, D., Kumar, B. and Armstrong, J. (to appear) "PANS: A portable navigation platform". *1995 IEEE Symposium on Intelligent Vehicles*.
- [4] Kim, K.I., Oh, S.Y., Lee, J.S., Han, J.H., and Lee, C.N. (1993) "An autonomous land vehicle: design concept and preliminary road test results". *1993 IEEE Symposium on Intelligent Vehicles*, pp. 146-151.
- [5] Kluge, K. and Thorpe, C. (1992) "Representation and recovery of road geometry in YARF", *1992 IEEE Symposium on Intelligent Vehicles*, pp. 114-119.
- [6] Marra, M., Dunlay, T.R. and Mathis, D. (1988) "Terrain classification using texture for the ALV." Martin Marietta Information and Communications Systems technical report 1007-10.
- [7] Nashman, M. and Schneiderman, H. (1993) "Real-time visual processing for autonomous driving". *1993 IEEE Symposium on Intelligent Vehicles*, pp. 373-378.
- [8] Pomerleau, D. A. (1994) *Neural Network Perception for Mobile Robot Guidance*, Kluwer Academic Publishing, Boston MA.
- [9] Rosenblum, M. and Davis, L.S. (1993) "The use of a radial basis function network for visual autonomous road following". *1993 IEEE Symposium on Intelligent Vehicles*, pp. 432-439.
- [10] Struck, G., Geisler, J., Laubenstein, F., Nagel, H. and Siegle, G. (1993) "Interaction between digital road map systems and trinocular autonomous driving". *1993 IEEE Symposium on Intelligent Vehicles*, pp. 461-465.
- [11] Want, J.S. and Knipling, R.R. (1993) *Single-Vehicle Roadway Departure Crashes: Problem Size Assessment and Statistical Description*. National Highway Traffic Safety Administration Technical Report DTNH-22-91-C-03121.
- [12] Zhang, J. and Nagel, H. (to appear) "Texture analysis and model-based road recognition for autonomous driving". To appear in *Journal of Computer Vision, Graphics and Image Processing*.