Vision-Based Neural Network Road and Intersection Detection and Traversal

Todd M. Jochem, Dean A. Pomerleau, and Charles E. Thorpe

The Robotics Institute, Carnegie Mellon University, Pittsburgh PA 15213

Abstract

The use of artificial neural networks in the domain of autonomous driving has produced promising results. ALVINN has shown that a neural system can drive a vehicle reliably and safely on many different types of roads, ranging from paved paths to interstate highways. The next step in the evolution of autonomous driving systems is to intelligently handle road junctions. In this paper, we present an addition to the basic ALVINN driving system which makes autonomous detection of roads and traversal of simple intersections possible. The addition is based on geometrically modelling the world, accurately imaging interesting parts of the scene using this model, and monitoring ALVINN's response to the created image.

1. Introduction

Much progress has been made toward solving the autonomous lane-keeping problem. Systems have been demonstrated which can drive robot vehicles at high speeds for long distances. Some systems use road models to determine where lane markings are expected[2][6][7], while others are based on artificial neural networks which learn the salient features required for driving on a particular road type[4][5][10].

The current challenge for vision-based navigation researchers is to build on the performance of the already developed lane-keeping systems, adding the ability to do higher level driving tasks. These tasks include actions such as lane changing, localization, and intersection detection and navigation. This papers examines the task of road and intersection detection and navigation.

ALVINN (Autonomous Land Vehicle In A Neural Network)[10] is the neural network based lane-keeping system upon which the work presented in this paper is based. Using simple color image preprocessing to create a grayscale input image and a 3 layer neural network architecture, ALVINN can learn in about 5 minutes, using back-propagation, the correct mapping from input image to output road location. This steering direction is used to control our testbed vehicle, a converted U.S. Army HMMWV called the Navlab 2. On this vehicle, ALVINN has driven at speeds up to 55 m.p.h. for 90 continuous miles.

The extended system, one which is capable of detecting roads and intersection, is called ALVINN VC (VC for Virtual Camera). ALVINN VC uses the robust road detection and confidence measurement capability of the core ALVINN system along with an artificial imaging sensor to reliably detect road segments which occur at locations other than immediately in front of the vehicle (and the camera.)

The imaging sensor that ALVINN VC uses is called a **virtual camera** and is described in detail in Section 2. Virtual cameras are the fundamental tool upon which the techniques presented in this paper are based. They provide a mechanism for determining the appropriateness of vehicle actions, but do not compromise the robust driving performance of the core ALVINN system.

2. The virtual camera

A virtual camera is simply an imaging sensor which can be placed at any location and orientation in the world reference frame. It creates artificial images using actual pixels imaged by a real camera that have been projected onto some world model. By knowing the location of both the actual and virtual camera, and by assuming a flat world model, accurate image reconstructions, called virtual images, can be created from the virtual camera location. Virtual camera views have been used by ALVINN VC to successfully navigate on all road types which the original ALVINN system performed.

An interesting issue that is a general theme of this paper is the ability of virtual cameras to merge neural systems with symbolic ones. Virtual cameras impose a geometric model on the neural system. In our case, the model is not a feature in an image, but rather a canonical image viewpoint which ALVINN VC can interpret. To ALVINN VC, the virtual camera is a sensing device. It is ALVINN VC's only link to the world in which it operates. ALVINN VC doesn't care where the virtual camera is located, only that it is producing images which are similar to those on which it was trained and can thus be used to locate the road. This interpretation may seem to trivialize ALVINN VC's functionality, but in reality, finding the road is what ALVINN VC is designed to do best. The virtual camera insures that the system gets images which will let it do its job to the best of its ability. The details of creating appro-

0-8186-7108-4/95 $4.00 © 1995 IEEE

priate virtual camera locations and interpreting the resulting output are left to other, higher level modules. So in essence, the virtual camera imposes a geometric model on ALVINN VC without it knowing, or even caring, about it. Used in conjunction with higher level modules, the model allows ALVINN VC to exhibit goal directed, intelligent behavior without compromising system performance.

3. Detection philosophy

There are three principles upon which road and intersection detection and navigation systems should be based. They are:

1. Detection and navigation should be data (image) driven.
2. Detection is signaled by the presence of features.
3. Road junctions should be traversed by actively tracking the road or intersection branch.

These principles and their relationship to ALVINN VC, as well as other road and intersection detection systems, are examined in greater detail in [3].

ALVINN VC uses a priori knowledge that specifies where and when appropriate virtual cameras should be created. The cameras are created relative to the vehicle and creation does not coincide with when the intersection is actually located, but rather somewhere before it occurs. Instead of "Now we are at the intersection, so look for its branches," the system deals with information like "Start looking for a single lane road." The virtual camera's location, and the network associated with each, is dependent upon the type of road that is expected to be encountered. When the road or intersection to be detected is present, the virtual cameras image it in a way that is meaningful to the system's neural networks. By continually monitoring the network's confidence for each virtual camera, the system can determine when the road or intersection is present.

ALVINN VC currently adheres to the first two principles of road junction detection and traversal mentioned earlier. Also, methods are under development that will allow ALVINN VC to actively track roads using a combination of active camera control and intelligently placed virtual cameras.

4. Other systems

Several other groups have built systems [1][8][10][11] to study the road and intersection detection problem. Many of them have adopted a data directed approach, but many also rely on the absence of features rather than the presence of them to indicate when a road or intersection is present. Few use active camera control. A complete summary of these system can be found in [3].

5. Experimental results

A series of experiments was conducted to assess the usefulness of virtual cameras for autonomously detecting roads and intersections. All of the experiments described were performed on the Navlab 2. The experimental site was a single lane paved path near the Carnegie Mellon campus. The path was unlined with grass on either side and was 3.1 meters wide.

The first experiment was very simple and designed to assess the basic ability of virtual cameras to create images which were usable by the system. For this, a single virtual view was used to detect an upcoming road and to navigate onto it. The second experiment was more challenging - virtual cameras were used to not only keep the vehicle on the road, but to also detect an upcoming 'Y' intersection. After the intersection was detected, the system used higher level information to choose the appropriate fork to follow.

5.1 Road detection experiment

This experiment was designed to test the system's robustness for detecting and navigating onto roads. In this experiment, the system had information that the vehicle was approaching a road perpendicularly. Its job was to detect the road, drive the vehicle onto it, and then continue operating in a normal autonomous driving mode. The vehicle was not on another road as it approached the road to be detected. This scenario could corresponds to ending a cross country navigation mission and acquiring a road to begin autonomous road following.

Initially, the vehicle was positioned approximately 35 meters off of the road which was to be detected, and aligned perpendicularly to it. A virtual view was created that was rotated 90 degrees from the direction of vehicle travel. This view was placed 20 meters in front of the vehicle. See Figure 1. The vehicle was instructed to move along its current heading until the system detected the road. At this point, the system instructed the vehicle to turn appropriately based on the point specified by the neural network. Once the system had aligned the vehicle sufficiently with the road, it was instructed to begin road following.

5.2 Road detection

The ability to detect the upcoming road was the first and most important requirement of the system. To accomplish this, every 0.3 second as the vehicle approached the road (at a speed of about 5 m.p.h.), a virtual image was created and passed to ALVINN VC's neural network. The network produced an output vector, interpreted as a point on the road to drive over, and a confidence value using the

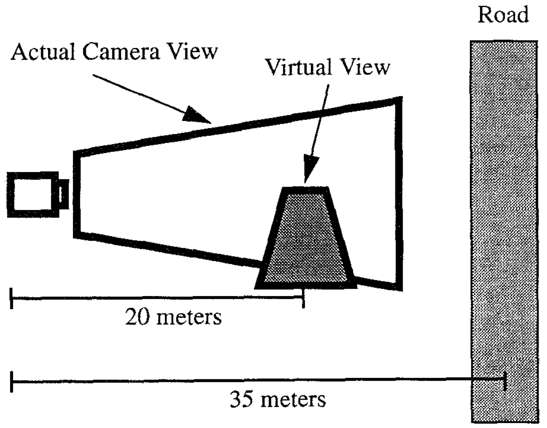


Figure 1

Input Reconstruction Reliability Estimation (IRRE) metric. This metric is described in greater detail in Section 5.3. To determine when the system had actually located the road, the IRRE metric was monitored. When this metric increased above a user defined threshold value, which was typically 0.8 (out of 1.0), ALVINN VC reported that it had located the road.

5.3 Application of IRRE to road detection

IRRE is a measure of the familiarity of the input image to the neural network. In IRRE, the network's internal representation is used to reconstruct, on a set of output units, the input pattern being presented. The more closely the reconstructed input matches the actual input, the more familiar the input and hence the more reliable the network's response.

The network is trained using backpropagation to both produce the correct steering response on the steering output units and to reconstruct the input image as accurately as possible on the reconstruction outputs.

During testing, images are presented to the network and activation is propagated forward to produce a steering response and a reconstructed input image. The reliability of the steering response is estimated by computing the correlation coefficient between the activation levels of units in the actual input image and the reconstructed input image. The higher the correlation between the two images, the more reliable the network's steering response is estimated to be[10].

Using the IRRE metric to indicate when roads are present in the input virtual image assumes that it will be low for images which do not contain roads and distinctly higher for those that do. For this assumption to hold, two things must occur. First, the system's neural network must not be able to accurately reconstruct images which do not contain roads, leading to a low IRRE measure. Second, images created by the virtual camera when a road is present must look sufficiently similar to ones seen during training, thus leading to an accurate reconstruction and a high IRRE response.

To test these assumptions several image were taken at various distances from the road as the vehicle approached. In each of these images, the location of the virtual camera was moved so that it imaged areas between the vehicle and the road, on the road, and past the road. Specifically, actual images were taken when the vehicle was at distances of 25, 20, 15, and 10 meters from the center of the road. Virtual camera images were created at 1 meter intervals on either side of the expected road location. For example, using the actual image taken 20 meters from the road center, virtual views were created every meter between the distances of 14 meters to 29 meters.

For each actual image, virtual camera images were created at intervals similar to those specified above and given to a network previously trained to drive on the one lane road. The output road location and the IRRE confidence metric were computed. The result of this experiment is shown in Figure 2. It shows the IRRE response with respect to the position of the virtual view for each actual image. For each actual image, the network's IRRE response clearly peaks near the expected road distance. As the virtual view moves closer to the road, the IRRE response increases, peaking when the virtual view is directly over the road. Response quickly falls again after the view passes over the road. For comparison, when ALVINN is driving on a familiar road, the IRRE response is typically between 0.70 and 0.95. The peaks in each IRRE curve actually occur about 2 meters past the actual road center. This is due to three things: a violation of the flat world assumption, errors in camera calibration, and improper initial alignment to the road.

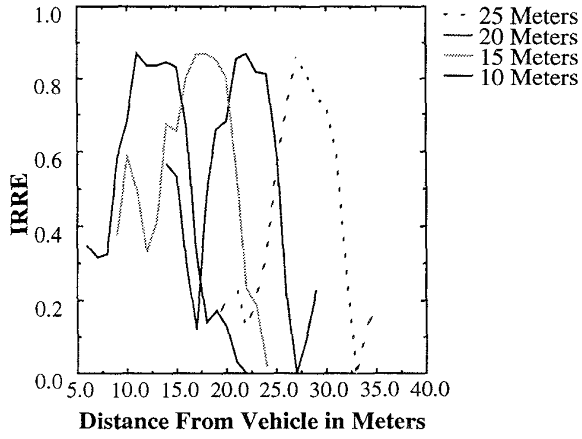
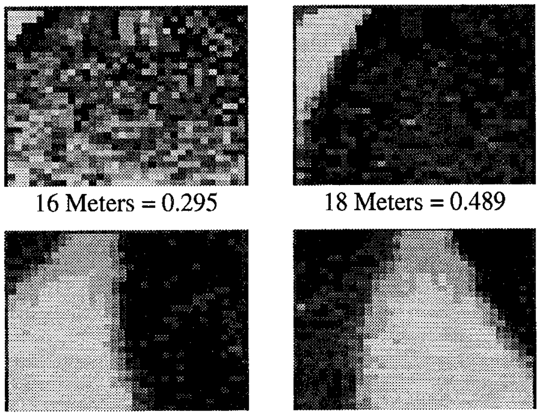


Figure 2

Figure 2 shows that both assumptions are basically correct - the IRRE response when the network is not being presented road images is low, and the IRRE response is high when the network is being presented accurately imaged virtual views.

The relationship between the input virtual image and the IRRE value associated with that image is better shown in Figure 3. The figure shows virtual images created at different distances in front of the vehicle along with the IRRE response they solicit. The images are all from an actual image that was taken when the vehicle was 20 meters from the road center. In the upper left image, the road is barely visible and, as expected, the IRRE response is very low. As the virtual view is moved forward, shown in the upper right and lower left images, it begins to image more of the road. The IRRE value increase correspondingly. The trend continues until the virtual view is centered over the road, as shown in the lower right image. At this location, the IRRE value peaks.



16 Meters = 0.295

18 Meters = 0.489

20 Meters = 0.680

22 Meters = 0.870

Figure 3

Each of the IRRE response curves shown in Figure 2 clearly indicate that a road is present at some distance in front of the vehicle. Because it is generally better to detect a road at a greater distance, it is desirable to know if accuracy in detection decreases as the distance from the vehicle to the road increases. Insight to this can be gained by transforming all of the curves from Figure 2 into the same reference frame. This can be done for each of the virtual views associated with a single actual image by subtracting the distance between the vehicle and the road center from the virtual camera location. This results in a coordinate system whose origin is at the center of the road. The result of transforming each of the response curves in Figure 2 into this coordinate frame is shown in Figure 4. This graph shows that detection accuracy, at least when approaching the road perpendicularly, does not degrade as distance from the road increases.

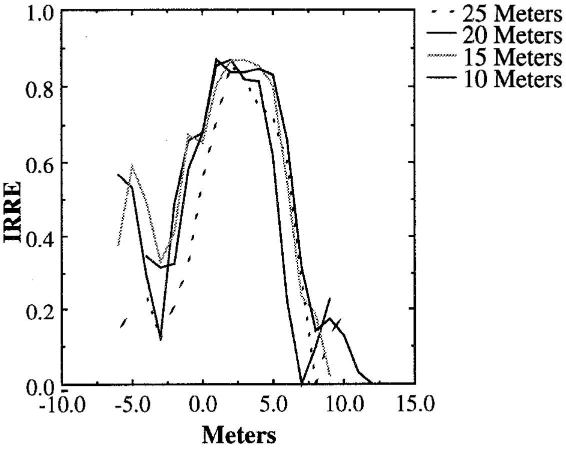


Figure 4

5.4 Alignment

While testing the detection phase of the system, it became clear that the problem would not be detecting the road, but rather driving onto it after it was detected. The next sections detail two algorithms used to drive the vehicle onto the road. The algorithms are presented in increasing order of robustness. The detection method described previously was used for finding the road for each method.

5.4.1. Simple road alignment

The first algorithm that was tested for moving the vehicle onto the road was to simply drive the vehicle over the point on the road which was specified by the system. For our vehicle, this meant that the center of the rear axle would pass over the specified road point. (The center of the rear axle is the origin of the vehicle coordinate system. Our point tracking algorithm uses this point as the location on the vehicle which should follow points to be tracked.)

The point tracking algorithm was able to reliably position the vehicle over the detected road point. The problem with this approach was that the vehicle heading was not matched with the road orientation. See Figure 5. Consequently, in many cases the vehicle was not able to begin road following after it had reached the road point because the road was no longer visible in the camera's field of view. One cause of this situation is that our point tracking algorithm, pure pursuit, does not attempt match desired and actual headings. But even if it did, the combination of the computed road point location relative to the vehicle origin and the minimum turn radius of the vehicle would prevent proper alignment to the road in some cases. Basically, the road position computed using the virtual view does not provide enough offset from the original

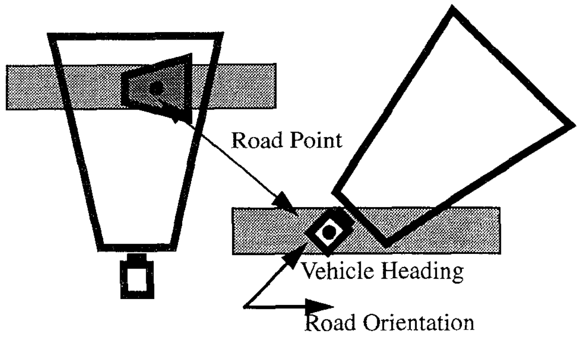


Figure 5

direction of travel to allow the necessary turn to be executed. The virtual view, and, in turn, the computed road position, is constrained by the actual camera - a large portion of the virtual view must be within the actual camera's field of view in order for realistic images to be created. This result suggests that another point on the road, further along it in the desired direction of travel, is needed.

5.4.2. Alignment by projecting along the road

To remedy the heading match problem encountered in the previous experiment, another point (P2) was created in addition to the network's output road location (P1). P2 was created using information about the orientation of the virtual view with respect to the vehicle. By doing this, it is assumed that the orientation of the virtual view is consistent with the expected road orientation. For example, when the virtual view is at a 90 degree angle with respect to the vehicle, P2 was created by projecting from P1 at an angle of 90 degrees from the line that runs forward from the vehicle origin. P2 is assumed to be on the road. The projection distance was typically 20 meters. See Figure 6.

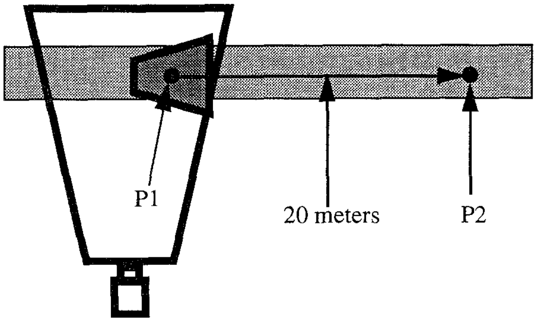


Figure 6

Whereas the first technique computed a single arc to drive to reach the road point, this technique requires more advanced interaction with the point tracking algorithm. Because the two points along with the vehicle location define a path to follow rather than just a point to drive over, other variables like the lookahead distance of the point tracker, the projection distance from P1 to P2, and the detection distance effect system performance. These parameters are discussed in detail in [3].

The selection of these parameters is not independent - changing one will likely require changing others in order to maintain system performance. This made developing a set of consistently usable parameters very difficult. In some trials, the vehicle turned smoothly onto the road and was able to begin road following. Other times, it turned too sharply and could not locate the road at all. In still other instances, it would cross over the road in its path to reach the projected road point.

There are two main disadvantages to this approach. The first is that it is not a trivial task to develop a correct set of point tracking parameters. Different parameters would likely be needed for every detection scenario and although it could be done, it would be a very large, brittle, and inelegant solution.

The second reason relates directly to determining P2. A large projection distance when computing P2 is desirable, but may not accurately model the road. A similar situation can happen even with small projection distances if the virtual view is not oriented exactly with the road. This occurs because ALVINN VC's neural network is trained on images which are created as if the vehicle was shifted and/or rotated from it true location. In the road detection scenario, this means that even if the road is not at the precise orientation defined by the virtual view, the network will respond with a high confidence. As a result, the road may not continue in the virtual view orientation and projecting to find P2 will yield a point off the road.

5.5 Intersection detection experiments

In this experiment, the goal was to drive along a single lane road, search for and detect a 'Y' intersection, and drive onto one fork or the other. See Figure 7. The central point of this experiment was to determine if intersections

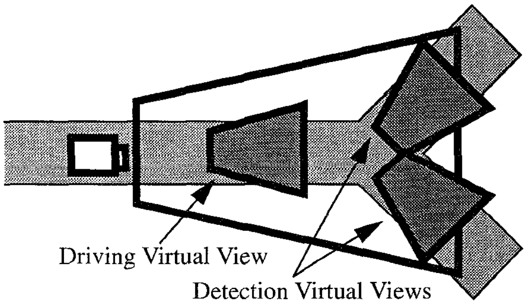


Figure 7

could be detected by extending the work done for detecting single roads. This experiment was more difficult than the previous road detection experiments for two reasons. First, it required that the system keep the vehicle on the road and at the same time look for the intersection branches. Second, it required that the system find two road branches rather than just one. Another factor adding to the difficulty of the scenario is that the intersection lies at the crest of a small hill - each of the road segments which meet at the intersection are inclined. This means that the flat world assumption is violated.

The road which the vehicle is travelling upon as well as each of the road branches are of the same type. Virtual views were created 9 meters in front of the vehicle. The view which was used to search for the left fork was angled 25 degrees to the left of straight ahead. The one used to search for the right fork was 20 degree right of straight ahead. Because of the geometry of the situation, the IRRE threshold value, which both virtual images were required to exceed, on a single actual image was lowered to 0.70. The experiment was conducted several times, with the results from each being similar to those of the single road case. The system was able to drive the vehicle at low speeds (5 m.p.h.) and detect each of the road branches. Although not as pronounced as in the single road detection case presented earlier, the system still had problems navigating onto either branch.

6. Conclusions and future work

Clearly, there is much work left to be done to robustly detect all roads and intersections. This paper presents a vision based approach which uses the core of a robust neural network road follower to accurately detect single lane, unlined roads. It is reasonable to assume that the detection method is directly extendable to any road type which the base neural network can learn to drive on. If this assumption is, in fact, found to be true, this system will have an advantage over other road and intersection detection systems which require the researcher to program in new detection methods when new road types are encountered.

Currently, the weak link of the system is its ability to navigate road junctions once they are found. We are investigating active camera control methods to address this problem.

Finally, the results presented were from a real, but fairly constrained environment. A robust road and intersection detection system must be able operate in more challenging environments - on typical city streets, with other cars, and with more extensive interaction with higher level knowledge. These areas are also actively being pursued.

7. Acknowledgments

This research was partly sponsored by DARPA, under contracts "Perception for Outdoor Navigation" (contract number DACA76-89-C-0014, monitored by the US Army Topographic Engineering Center) and "Unmanned Ground Vehicle System" (contract number DAAE07-90-C-R059, monitored by TACOM) as well as a DARPA Research Assistantship in Parallel Processing administered by the Institute for Advanced Computer Studies, University of Maryland.

8. References

[1] Crisman, J. D. *Color Vision for the Detection of Unstructured Roads and Intersections.* Ph.D. dissertation, Carnegie Mellon University, May, 1990.

[2] Dichmanns, E.D. and Zapp, A. "Autonomous high speed road vehicle guidance by computer vision." *Proceedings of the 10th World Congress on Automatic Control,* Vol. 4, Munich, West Germany, 1987.

[3] Jochem, T. Pomerleau D., and Thorpe, C. "Initial Results in Vision Based Road and Intersection Detection and Traversal," CMU Technical Report CMU-RI-TR-95-21, April, 1995.

[4] Jochem, T. Pomerleau, D., Thorpe, C. "MANIAC: A Next Generation Neurally Based Autonomous Road Follower," *Intelligent Autonomous Systems-3,* February 1993, Pittsburgh, PA, USA.

[5] Jochem, T. and Baluja, S. "Massively Parallel, Adaptive, Color Image Processing for Autonomous Road Following," in *Massively Parallel Artificial Intelligence*, Kitano and Hendler (ed), AAAI Press, 1994.

[6] Kenue, S.K. (1989) "Lanelok: Detection of lane boundaries and vehicle tracking using image-processing techniques," *SPIE Conference on Aerospace Sensing, Mobile Robots IV,* Nov. 1989.

[7] Kluge, K. *YARF: An Open Ended Framework for Robot Road Following.* Ph.D. dissertation, School of Computer Science, Carnegie Mellon University, February 1993.

[8] Kluge, K and Thorpe C. "Intersection Detection in the YARF Road Following System," *Intelligent Autonomous Systems 3,* February 1993, Pittsburgh, PA, USA.

[9] Pomerleau, D.A. "Efficient Training of Artificial Neural Networks for Autonomous Navigation," *Neural Computation 3:1*, Terrence Sejnowski (Ed).

[10] Pomerleau, D. *A. Neural Network Perception for Mobile Robot Guidance.* Ph.D. dissertation, Carnegie Mellon University, February, 1992.

[11] Rossle, S., Kruger, V., and Gengenbach. "Real-Time Vision-Based Intersection Detection for a Driver's Warning Assistant," *Intelligent Vehicle '93,* July 1993, Tokyo, Japan.