Performance evaluation of road detection and following algorithms

Tsai Hong, Aya Takeuchi, Mike Foedisch, Michael Shneier
National Institute of Standards and Technology
Gaithersburg, MD 20899

ABSTRACT

We describe a methodology for evaluating algorithms to provide quantitative information about how well road detection and road following algorithms perform. The approach relies on generating a set of standard data sets annotated with ground truth. We evaluate the algorithms used to detect roads by comparing the output of the algorithms with ground truth, which we obtain by having humans annotate the data sets used to test the algorithms. Ground truth annotations are acquired from more than one person to reduce systematic errors. Results are quantified by looking at false positive and false negative regions of the image sequences when compared with the ground truth. We describe the evaluation of a number of variants of a road detection system based on neural networks.

1. INTRODUCTION

Performance evaluation is not commonly practiced in the computer vision community. Periodically, efforts are made to persuade researchers to provide performance evaluations that can be substantiated, but only a few take up this challenge. As a result, performance evaluation is ad hoc in general and quite frequently completely absent from research papers.

In Europe, a number of formal programs have been developed that address performance evaluation of vision algorithms. Of these, ECVnet, an association of European vision researchers, had a subcommittee on Benchmarking and Performance Measures[1], although it now appears to be defunct. The German Association for Pattern Recognition (DAGM) established a Working Group on "Quality Evaluation of Pattern Recognition Algorithms", but it, too appears inactive[2]. The International Association for Pattern Recognition has a Technical Committee on Benchmarking & Software, which organizes performance competitions comparing algorithms for particular applications, such as fingerprint identification and document analysis.[3] There have also been a number of workshops on performance characterization and benchmarking of vision systems.

A number of publications address the issue of how to evaluate the performance of vision algorithms, and show a few examples of careful evaluations of particular algorithms or classes of algorithms. Approaches to performance evaluation can be classified into the following general categories, recognizing that more than one approach may be used in an evaluation.

Comparative Here an algorithm may be compared with others that attempt to address the same image-processing task, or its performance may be compared to "ground truth," or perhaps to human performance.[4, 5, 6, 7, 8, 9]

Analytic The theory behind the algorithm is examined to try to determine the limits to its operation. The computational complexity may be derived, or theoretical optimality may be determined under certain constraints. Frequently, the approach makes simplifying assumptions to make the analysis feasible.[10, 11, 12, 13]

Performance The way the algorithm actually performs on test data is measured and execution times with different parameters may be reported.[14, 15, 16]

Appropriateness to Task The algorithm is shown in the context of a particular application, and the constraints of the task are used to justify the selection of the particular algorithm. The performance of the task as a whole is taken as the evaluation of the algorithm.[17, 18]

More informal measures include generality and acceptance. Perhaps the only real performance evaluation measure in common use is longevity. Algorithms that are accepted widely and implemented by many people for different applications can be considered good performers.

A large number of papers report excellent performance of their algorithms, based on small data sets. The success of the

Face Recognition Technology (FERET) program[9] has inspired us to take up the challenge of producing a large database of ground truth for the domain of mobile robotics. In this domain, sensors are mounted on board a moving vehicle, and the algorithms are constrained to run in real time (i.e., fast enough to provide data to control the vehicle). The ground truth that we provide is more extensive than is typically available. Human interpretations provide the ground truth which covers a large number of images. We describe our approach to generating ground truth and demonstrate performance evaluations in the domain of road detection and tracking. The data sets are used to evaluate performance of algorithms objectively by comparing the output of the algorithms to the expected results derived from the ground truth. Given a large number of ground truth data sets from different environments, statistical evaluations are possible as well as the robust assessment of performance of algorithms.

The main goal of this work is to make our test data and ground truth available for general use, with the hope that it will lead to rapid and significant development of perception algorithms for autonomous mobility. In order to validate the approach we use the data sets to evaluate our own algorithms developed for the domain of road detection and tracking.

2. GROUND TRUTH

To generate ground truth for road detection and tracking, we start with a large collection of video sequences taken with a camera mounted on a vehicle driven over roads of all types, including dirt and gravel roads, suburban streets, and highways. These sequences are part of a large repository of sensor data (close to a terabyte) developed over the past few years as part of the Army's Demo III program[19] and the Defense Advanced Research Projects Agency (DARPA) Mobile Autonomous Robot Software (MARS) program. We have developed a semi-automated approach to annotating the data sets to create ground truth, which we define as segmentation of the images in the video sequence by a human. An application has been developed that allows the ground truth to be extracted and stored in a database for later use.

A human user uses a graphical user interface to select a video sequence. A frame from the sequence is displayed and a tool is applied to segment the image into road and non-road regions (Figure 1). This requires the user to outline the road region by selecting points on the boundary between road and non-road. When the user is satisfied with the segmentation, the bounding segments are stored in a table in a database (we use MySQL[20†]) and the process is repeated on the next frame. The table stores only the coordinates of the outlined region, which is sufficient to reconstruct the region for use in evaluation. We plan to improve the application so that it can automatically segment successive frames, with the human viewing the results and stepping in to re-initialize the process when the automatic segmentation begins to drift.

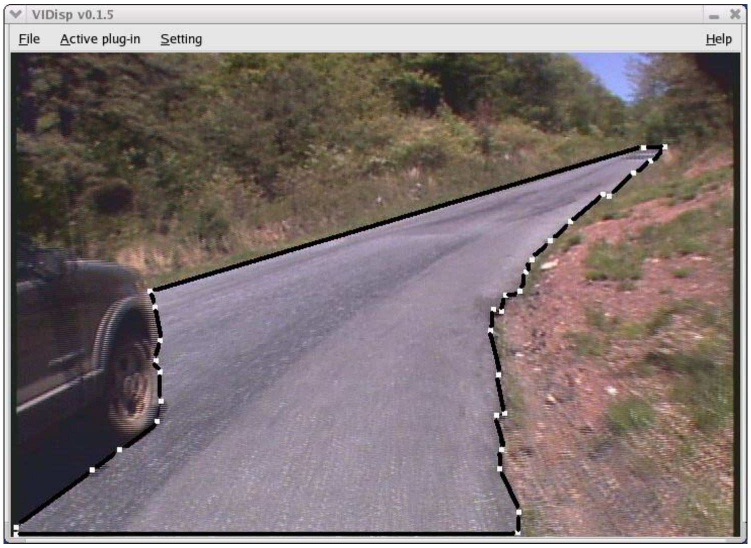

Figure 1. A view of the tool for annotating ground truth showing the road region outlined in a video frame.

[†] Certain commercial equipment, instruments, or materials are identified in this paper in order to adequately specify the experimental procedure. Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the materials or equipment identified are necessarily best for the purpose.

The database stores the results of the segmentation or of multiple segmentations by different users. The segmentations are used to evaluate algorithms by comparing their output to the ground truth.

3. ALGORITHM EVALUATION

Evaluating an algorithm is fairly straightforward. The algorithm is applied to a video sequence for which ground truth is available, and the output is stored as a new video sequence. The evaluation then involves taking successive pairs of frames from the ground truth and the algorithm output and comparing them pixel by pixel. Where the ground truth differs from the algorithm's output, an error is registered. This can be a false positive (a non-road pixel labeled as road) or a false negative (a road pixel labeled as not road). Numbers of false positives and false negatives are stored separately, and are accumulated over all frames. Also output is a new video, with each frame showing the locations of the false positives and negatives (Figure 4). This is helpful because the locations of the errors may be of very little concern (e.g., if they lie along the road edges) or may be unacceptable if they occur far away from the road boundaries.

3.1 Road detection algorithms

We describe the performance evaluation of three variants of a neural network-based road detection algorithm. Neural network training requires that regions be identified as road and non-road. To accomplish this, the algorithms make use of six windows, three that are on the road and three that are on the background. The differences between the variants of the algorithm are in how the windows are placed over the image and when the training of the network is carried out.

The first variant makes use of windows of fixed size and position. When entering a new road, the first few seconds worth of data are used to train the neural network. Three windows are placed over the road region. Features taken from these windows will be labeled as road. Similarly, three windows are placed over non-road regions and result in similar regions being labeled as non road (Figure 2). The windows remain fixed in the same locations for all frames. Once the network has been trained using these initial frames of data, it is used to recognize roads in all subsequent frames.

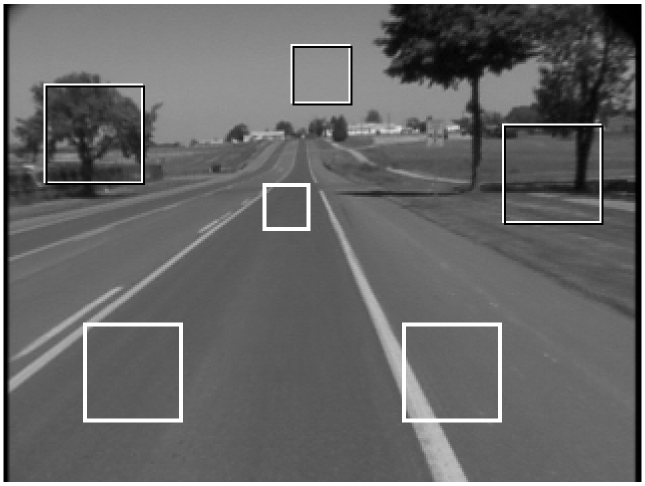

Figure 2. Six fixed windows placed over a sample frame to train a neural network.

The second variant is similar to the first, except that the network is re-trained periodically. It is frequently the case that the appearance of the road surface changes over time (e.g., the surface might change from asphalt to concrete, or a sunlit road might give way to a shaded one). To make the algorithm robust to such events, the network is re-trained by taking features from the road and non-road windows at a fixed frequency as the vehicle moves along the road.

Both variants of the algorithm make the assumption that the windows that represent road regions will continue to represent such regions in subsequent frames, and similarly for the non-road windows. This assumption is frequently violated, for example, when there is a curve in the road (Figure 3). To overcome this problem, a third variant of the algorithm was developed in which the positions of the road windows are not fixed. In this variant, the windows that represent the road region may be moved to ensure that they remain entirely on the road. The non-road windows are not moved, but individual windows may be switched off when they overlap the road.

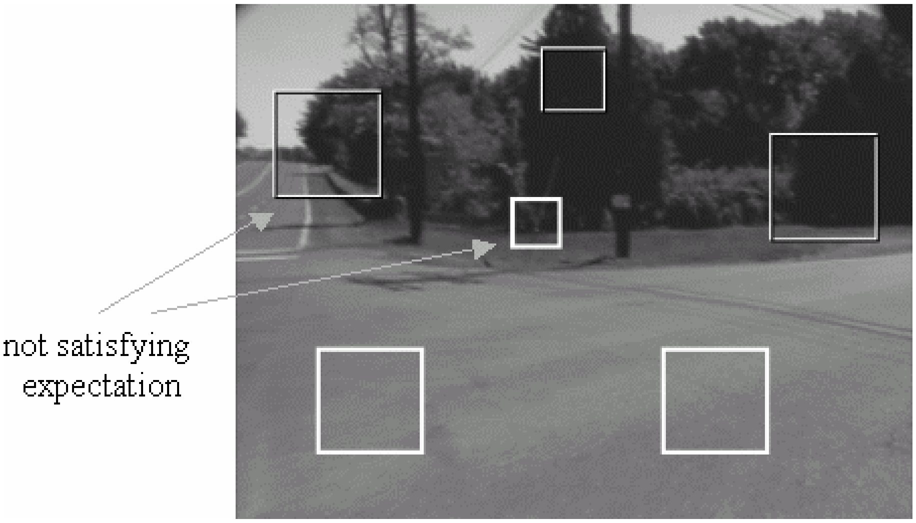

Figure 3. Example of windows violating the assumptions due to a change in the curvature of the road.

3.2 Performance evaluation

We conducted experiments to evaluate the performance of the three variants of the road detection algorithm. Each algorithm was applied to a set of data for which ground truth had been manually defined using the tool described in section 2. Results were gathered on a frame-by-frame basis for each of the algorithms by comparing the output of the algorithm with the ground truth pixel by pixel. There are two possibilities. Either both values agree with each other (e.g. both label a region as road or non-road) or they don't. When they disagree, there are two cases: 1) the road detection algorithm claims a point to be road but the ground truth specifies the point as non-road – in this case we speak of "false positive" classification; 2) the road detection algorithm claims a point to be non-road while the ground truth says otherwise – in this case we speak of "false negative" classification.

Figure 4 depicts the classification results of the three variants of the road detection algorithm applied to a single frame. Black dots show areas which were classified as being non-road, white dots show areas classified as road. Black and white dots visualize areas where the road detection algorithm agrees with the ground truth, whereas black blocks show road-areas erroneously classified as non-road (false negatives) and white blocks show non-road areas classified as road (false positives).

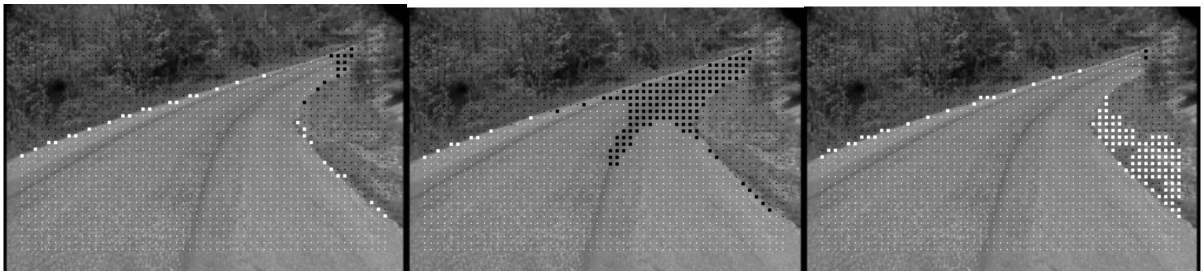

Figure 4. Results of the three variants of the algorithm (left to right) showing false positives and false negatives.

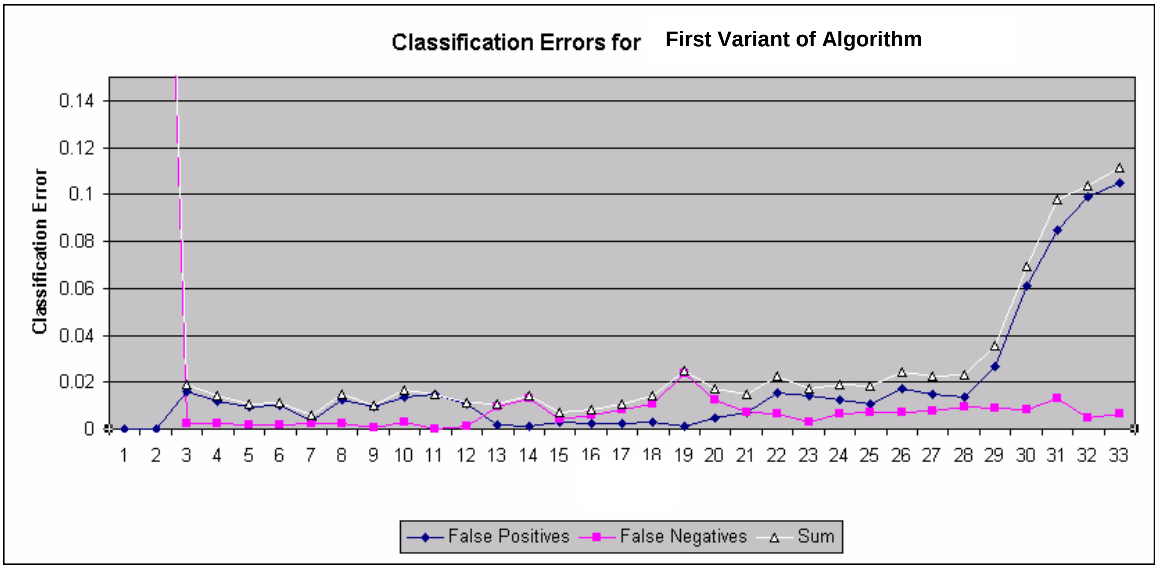

Figure 5. Graph showing classification results for every 25th frame of a video sequence.

Figure 5 shows the classification results for every 25th frame of the analyzed video sequence, using the first variant of the algorithm. The three curves describe the false positive and false negative classification rate as well as the sum of both. The first frames illustrate a characteristic of the applied road detection algorithm, namely a low false positive and a high false negative rate. Initially, when the algorithm is applied to the video sequence, the neural network has not been trained, so the distinction between road and non-road areas for this particular scene is unknown. In this case, all of the image areas were classified as non-road, which leads to the high false negative rate in the beginning. After a short adaptation phase the algorithm trained a neural network to classify image areas correctly, leading to a low false negative rate.

The graphs make it easy to visualize a road detection algorithm's classification results, and let us pinpoint situations where the algorithm performs well or badly.

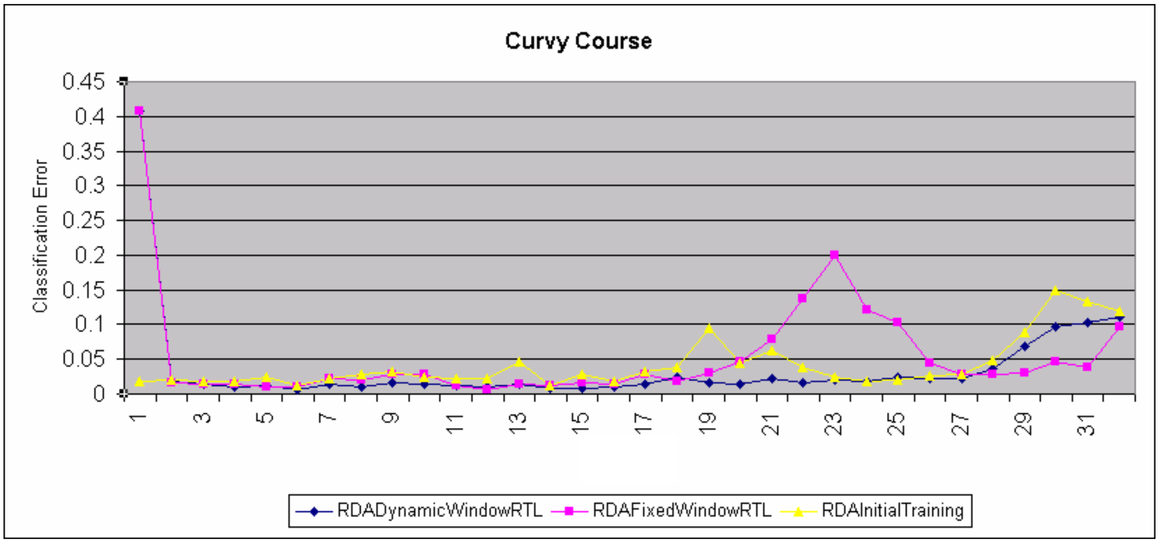

Figure 6. Sum of false positives and false negatives for the three classification algorithms. Once again, every 25th frame is shown.

Figure 6 depicts the classification results (sum of false positives and false negatives only) of the three road detection algorithms for every 25th frame of a video sequence. While Figure 5 allowed the analysis of classification results for a single algorithm in detail, Figure 6 supports the comparison of several algorithms' performance on the same data. This visualization allows us to pinpoint essential differences in the way the algorithms behave, e.g. in the beginning of the course and around the image numbered 23 in the graph. While the differences in the beginning are caused by different initialization approaches, the bad results of the "Fixed Window" algorithm around image number 23 were caused by inherent problems in the way the algorithm deals with curvy roads and turnings that result in the fixed windows being incorrectly placed.

4. CONCLUSIONS

This paper has described our approach to performance evaluation of road detection algorithms, which involves manually generating ground truth and then comparing the output of the algorithm under test with the "true" result. The ground truth developed for this purpose is useful for performance evaluation of other algorithms also. Our repository of image sequences is large and includes ground truth for features other than roads. The application to three variants of a neural-network based road detection algorithm illustrates the analysis that can be done to help understand the way the algorithms work and the advantages and disadvantages of different algorithms. Quantitative differences can be measured in the false positive and false negative results of each algorithm. It is important, however, to examine the output of the evaluation visually because one algorithm might appear to have worse performance than another, but its errors may be visually more innocuous. In our example of road detection, an algorithm may have a large number of errors, but if all the errors lie on the boundary between road and non-road, it might be preferable to an algorithm with a lower error rate whose errors are distributed widely over the image.

Performance evaluation is particularly important when sensor processing is being used in mission critical applications. For our work in mobile robotics, we plan to continue to evaluate our algorithms so as to be able to characterize the situations in which they work and identify situations where more development is required.

Acknowledgements

We are grateful for the support of the Army Research Laboratory and the DARPA MARS project.

REFERENCES

1. Courtney, P., Benchmarking and Performance Evaluation , http://www-prima.inrialpes.fr/ECVNet/benchmarking.html, Mar.,1998.

2. Faber, A., Quality Characteristics of Pattern Recognition Algorithms, http://www.dagm.de/DAGM/ag/wg.html, May,1998.

3. Lucas, S., IAPR TC-5 Benchmarking and Software, http://algoval.essex.ac.uk/tc5/Introduction.html, 2003.

4. Bowyer K., Kranenburg, C., and Dougherty, S., "Edge Detector Evaluation Using Empirical ROC Curves," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 354-359, IEEE, Los Alamitos, CA, 1999.

5. Nguyen, T. B. and Zhou, D.,"Contextual and Non-Contextual Performance Evaluation of Edge Detectors," *Pattern Recognition Letters* **21**, 805-816,2000.

6. Matthies, L., Litwin, T., Owens, K., Murphy, K, Coombs, D., Gilsinn, J., Hong, T., Legowik, S., Nashman, M., and Yoshimi, B., "Performance Evaluation of UGV Obstacle Detection with CCD/FLIR Stereo Vision and LADAR," *IEEE Workshop on Perception for Mobile Agents*, Santa Clara, CA, 1998.

7. Shufelt, J. A.,"Performance Evaluation and Analysis of Monocular Building Extraction From Aerial Imagery," *IEEE Transaction on Pattern Analysis and Machine Intelligence* **21**, 311-326,1999.

8. Wiedemann C., Heipke, C., Mayer, M., and Jamet, O., "Empirical Evaluation of Automatically Extracted Road Axes." *Empirical Evaluation Techniques in Computer Vision*, 172-187, 1998.

9. Phillips, P. J., Moon, H., Rizvu, S. A., and Rauss, P.,"The FERET Evaluation Methodology for Face-Recognition Algorithms," *IEEE Transaction on Pattern Analysis and Machine Intelligence* **22**, 2000.

10. Cho, K., Meer, P, and Cabrera, J, "Performance Assessment Through Bootstrap," *IEEE Transaction on Pattern Analysis and Machine Intelligence* **19**, 1185-1198,1997.

11. Courtney, P., Thacker, N., and Clark, A. F.,"Algorithmic Modelling for Performance Evaluation," *Machine Vision and Applications* **9**, 219-228, 1997.

12. Kiryati, N., Kälviäinen, H., and Alaoutinen, S., "Randomized or Probabilistic Hough Transform: Unified Performance Evaluation," *Pattern Recognition Letters* **21**, 1157-1164,2000.

13. Haralick, R., "Propagating Covariance In Computer Vision," *Proceedings of the ECCV Workshop on Performance Characteristics of Vision Algorithms*, Cambridge, England, 1996.

14. Pissaloux, E. E.,"Toward an image segmentation benchmark for evaluation of vision systems," *Journal of Electronic Imaging* **10**, 203-212, 2001.

15. Min, J., Powell, M. W., and Bowyer K., "Automated performance evaluation of range image segmentation," *Fifth IEEE Workshop on Applications of Computer Vision*163-168, IEEE, Palm Springs, CA, 2000.

16. Coutre, S. C, Evens, M. W., and Armato II, S. G., "Performance Evaluation of Image Registration," *Proceedings of the 22nd Annual EMBS International Conference,* 3140-3143, IEEE, Chicago, IL, 2000.

17. Shin, M. C., Goldgof, D., and Bowyer, K. W., "Objective Comparison Methodology of Edge Detection Algorithms Using a Structure From Motion Task, "*Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, Santa Barbara, CA, 1998.

18. Moon, H., Chellappa, R., and Rosenfeld, A., "Performance Analysis of a simple Vehicle Detection Algorithm," *Image and Vision Computing* **20**, 1-13, 2002.

19. Shoemaker, C. M. and Bornstein, J. A., "The Demo3 UGV Program: A Testbed for Autonomous Navigation Research," *Proceedings of the IEEE International Symposium on Intelligent Control*, Gaithersburg, MD, 1998.

20. MySQL AB, MySQL, http://www.mysql.com/, 2004.