Towards self driving: Single camera navigation localisation
Final report

Michael McDonnell*

UNSW Canberra at ADFA.

ABSTRACT

*CAPT, School of Engineering and Information Technology, ZEIT4902

Contents

I. Introduction

A. Terminology

The following abbreviations and definitions are used throughout this report:

- **Computer Vision (CV)**. Techniques to allow machines to 'see'; processing visual images of the world and deriving understanding.
- **Digital Image Processing (DIP)**. Use of computer algorithms to perform processing on digital images.
- **Navigation localisation**. Translating an overall navigation route into a local navigation goal.
- **Simulation**. The custom built autonomous vehicle simulation which provides sensor feed outputs.
- **External processing**. In the context of this report two programs are discussed, the simulation and 'external processing'. External processing refers to the standalone code which performs the computational calculations for navigation localisation.
- **Simulation tick**. One period of simulation processing. This is aligned to a specified update frequency that the external processing is running at 'in simulation'.
- **Interprocess Communication (IPC)**. The passage of data from simulation to external processes.
- **Open Street Maps (OSM)**. 'A collaborative project to create a free editable map of the world' (OSM, 2019). Comparable to Google maps.
- **Polyline**. A line defined by node coordinates. The line is drawn through all nodes in order, starting at the first and terminating at the last.

B. Project aim

The aim of this project is to investigate localising navigation data from a GPS feed to the observed road via a vehicle mounted camera. Mapping service GPS route data is often held as polylines which represents a road as a series of connected points or nodes (Google, 2018), (OSM, 2019). The approach for positional localisation is to use the vehicle GPS position as an approximate input location mapped to the closest point on a route. Detected road features are then used to determine an accurate position of the vehicle and identify the navigation route on forward facing video feed. This sets the conditions for autonomous control of the vehicle based on a programmed navigation route.

C. Scope and Deliverables

The scope of the project is deliberately kept constrained initially. This is to focus on the specific problem of localising a navigation route without losing development effort to supporting elements. The scope and deliverables have been identified as follows:

- The solution must be able to reconcile GPS and CV data to identify the current location and required direction to travel through intersections based on a navigation route.
- **Limitation of road complexity.** There is a requirement for road/lane detection as part of this project (to marry up with the GPS polyline data) however optimised road detection is not the main focus of the project. Further as the project purely uses the output data of lane detection, it can be considered a 'black box' and implementations can be swapped out as more advanced options are identified. The initial limitations on scope of road detection includes:
  - Limit road detection to easily detectable road surface.
  - Limit roads to single lane.
  - All roads considered will be of similar local colour and type.
- **Simulation deliverable requirements.** In addition to providing data for this project the intent is for the simulation to be held as an asset within SEIT for use in subsequent student projects in this area. The basic requirements for the simulation are:
  - Ability to provide 3D video feed of simulated driving to external program.
  - Support simulated GPS tracking data.

– Support simulation of GPS route guidance.

II. Project management

??????

III. SOLUTION overview

TODO: Reword title TODO: Discuss the basic overview, relate to human?

- ????
  - **TODO? Explain - Inverse perspective mapping and camera lens distortion correction**
- Probabilistic road surface detection
  - **TODO? Explain - Rolling average histogram for road surface estimation**
- Local orientation to road edges
  - **TODO? Explain - Hough transform (IS THIS NEEDED?)**
- Route feature matching
  - **TODO? Explain - Matching piecewise model to detected road surface by masking**
- Optical flow tracking
  - **TODO? Explain - Using image 'flow' to track feature positions for subsequent frame matching and cornering**
- Driving path curve matching
  - **TODO? Explain - Bezier curves based off feature points**

A. ??? IPM, camera correction

Preliminaries

B. Probabilistic road surface detection

TODO: Define area of interest in camera (IMAGE) TODO: Maintain rolling average of area of interest and calculate histogram of this region TODO: Histogram backprojection (SIMULATION AND/OR LIVE IMAGE EXMAPLES, LIVE OF VARYING ROAD TYPES) TODO: Thresholding vs raw probabilities??? Raw probabilities with 'upgraded probabilities based on thresholding???

C. Local road edge orientation

TODO: Hough? IS THIS NEEDED?!?!?

D. Route feature matching

TODO: 'Feature' definition TODO: Feature mask creation TODO: Piecewise feature matching TODO: Feature center point determination

E. Optical flow tracking

TODO: Optical flow theory TODO: Frame by frame tracking - estimation of feature center point TODO: Confirm using route feature matching as per section D

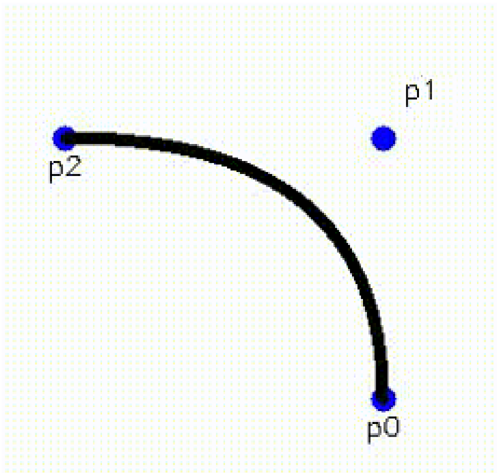

Figure 1. Quadratic Bezier curve from control points (p_0, p_1 and p_2)

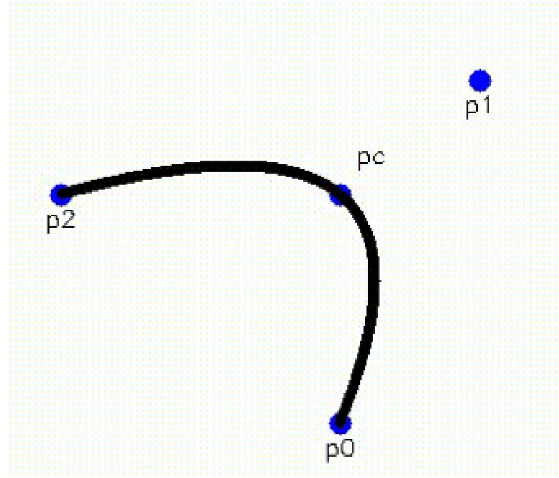

Figure 2. Quadratic Bezier curve from control points (p_0, p_1 and p_2) constrained to pass through p_c

F. Driving path curve matching

TODO: DO we need to create control points near intersection for p0 and p2 or do the poly line points work ok? I imagine we will need to artificially create these entry/exit points from the feature!!! This can be done simply by moving X meters along the approach/exit line to determine entry/exit control points

The real challenge in this problem is that most route features require more than a simple maintenance of central position such as turning at an intersection. In order to effectively identify a path through a route feature, a driving path is developed using bezier curves.

1. Bezier curve theory

TODO: ADD THIS AS APPENDIX??!? A Bezier curve is a parametric curve defined by a series of control points with the position on the curve defined by a variable t ranging between 0 and 1, corresponding to the start and the end of the curve. Quadratic Bezier curves consist of three control points (p_0, p_1 and p_2) resulting in a quadratic equation with respect to the variable t. **TODO: Keep going**

TODO: IS THIS MID POINT INTERPOLATION REQUIRED? OR DOES SETTING 'IDEAL' CONTROL POINTS FIX IT? The formula for the basic Quadratic Bezier curve is outlined as equation 1. This will result in a curve as per the curve in figure 1. A shortfall with this approach is it will result in 'cutting' of the corner as the point p_1 is the center point of the feature. This can be addressed by using what was p_1 as the central point p_c to calculate a middle control point to force the curve through this point. If a constraint is added that $\mathbf{B}(t) = p_c$ for some value of t, it is then possible to solve for the required p_1 to fit this constraint. The free parameter in this instance is the value of t where the curve passes through p_c. Setting $t = 0.5$ and solving for p_c will set the curve to pass through p_c at the mid point which was deemed suitable in this instance. The equation to force the p_c constraint is included as equation 2 and the result of forcing the constraint $\mathbf{B}(0.5) = p_c$ is visualised in figure 2.

$$\mathbf{B}(t) = (1-t)^2 p_0 + 2t(1-t)p_1 + t^2 p_2 \tag{1}$$

$$p_1 = (p_c - t^2 p_0 - t^2 p_2)/t \tag{2}$$

$$p_{mid} = (1-a)p_1 + a p_c \tag{3}$$

It is clear the developed both path curves in both figures 1 and 2 are not an ideal solution. As a result the final approach was to use a weighted interpolation between p_c and p_1 where a is a weighting between 0 and 1 for p_c as the middle control point position; when $a = 0$, p_c will have full weighting over the calculated p_1 as the middle control point. **TODO: a = 0.5**

2. Path matching

Once the path is developed, the approach used in section D is used to match the developed path to the detected road surface. The developed path is then used for subsequent matching and to facilitate driving control.

References

M. Aly. Real time detection of lane markers in urban streets. In *2008 IEEE Intelligent Vehicles Symposium*, pages 7–12, June 2008. doi: 10.1109/IVS.2008.4621152.

M. Bertozz, A. Broggi, and A. Fascioli. Stereo inverse perspective mapping: Theory and applications. *Image and Vision Computing*, 16:585–590, 06 1998. doi: 10.1016/S0262-8856(97)00093-0.

M. Bosse and R. Zlot. Map matching and data association for large-scale two-dimensional laser scan-based slam. *I. J. Robotic Res.*, 27:667–691, 06 2008. doi: 10.1177/0278364908091366.

J. D. Crisman and C. E. Thorpe. Color vision for road following. In *Vision and Navigation: The CMU Navlab, C. Thorpe (Ed*, pages 9–24. Kluwer Academic Publishers, 1988.

J. D. Crisman and C. E. Thorpe. Scarf: a color vision system that tracks roads and intersections. *IEEE Transactions on Robotics and Automation*, 9(1):49–58, Feb 1993. ISSN 1042-296X. doi: 10.1109/70.210794.

J. D. Crisman and C. E. Thorpe. Unscarf–a color vision system for the detection of unstructured roads. volume 3, pages 2496 – 2501 vol.3, 05 1991a. doi: 10.1109/ROBOT.1991.132000.

J. D. Crisman and C. E. Thorpe. Unscarf–a color vision system for the detection of unstructured roads. volume 3, pages 2496 – 2501 vol.3, 05 1991b. doi: 10.1109/ROBOT.1991.132000.

H. Dahlkamp, A. Kaehler, D. Stavens, S. Thrun, and G. R. Bradski. Self-supervised monocular road detection in desert terrain. In *Robotics: Science and Systems*, 2006.

R. O. Duda and P. E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM*, 15:11–15, 1972.

GM. General motors 2018 self-driving safety report, 2018. URL `https://www.gm.com/content/dam/company/docs/us/en/gmcom/gmsafetyreport.pdf`.

J. P. Gonzalez and U. Ozguner. Lane detection using histogram-based segmentation and decision trees. In *ITSC2000. 2000 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.00TH8493)*, pages 346–351, Oct 2000. doi: 10.1109/ITSC.2000.881084.

Google. Encoded polyline algorithm format (google maps documentation). Website, Dec. 2018. URL `https://developers.google.com/maps/documentation/utilities/polylinealgorithm`.

J. Illingworth and J. Kittler. A survey of the hough transform. *Comput. Vision Graph. Image Process.*, 44 (1):87–116, Aug. 1988. ISSN 0734-189X. doi: 10.1016/S0734-189X(88)80033-1. URL `http://dx.doi.org/10.1016/S0734-189X(88)80033-1`.

T. Jochem. Initial results in vision based road and intersection detection and traversal. 1995.

Jun Wang, Tao Mei, Bin Kong, and Hu Wei. An approach of lane detection based on inverse perspective mapping. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 35–38, Oct 2014. doi: 10.1109/ITSC.2014.6957662.

KIA. Kia sorento vehicle information, 2019. URL `https://www.kia.com/us/en/vehicle/sorento/2019`.

T. Litman. Autonomous vehicle implementation predictions: Implications for transport planning, Mar. 2019. URL `https://www.vtpi.org/avip.pdf`.

H. Mallot, H. Blthoff, J. Little, and S. Bohrer. Inverse perspective mapping simplifies optical flow computation and obstacle detection. *Biological cybernetics*, 64:177–85, 02 1991. doi: 10.1007/BF00201978.

D. Neven, B. De Brabandere, S. Georgoulis, M. Proesmans, and L. Van Gool. Towards end-to-end lane detection: an instance segmentation approach. pages 286–291, 06 2018. doi: 10.1109/IVS.2018.8500547.

NHTSA. Automated driving systems 2.0: A vision for safety. Online, Sept. 2017. URL `https://www.nhtsa.gov/sites/nhtsa.dot.gov/files/documents/13069a-ads2.0_090617_v9a_tag.pdf`.

G. L. Oliveira, W. Burgard, and T. Brox. Efficient deep models for monocular road segmentation. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4885–4891, Oct 2016. doi: 10.1109/IROS.2016.7759717.

OpenCV. Open Source Computer Vision Library, 2019. URL `https://opencv.org/about/`.

T. Ort, L. Paull, and D. Rus. Autonomous vehicle navigation in rural environments without detailed prior maps. pages 2040–2047, 05 2018. doi: 10.1109/ICRA.2018.8460519.

OSM. Osm file formats, Feb. 2019. URL `https://wiki.openstreetmap.org/wiki/OSM_file_formats`.

Qingmei Yang and Jianmin Sun. A location method for autonomous vehicle based on integrated gps/ins. In *2007 IEEE International Conference on Vehicular Electronics and Safety*, pages 1–4, Dec 2007. doi: 10.1109/ICVES.2007.4456376.

S. Tan, J. Dale, A. Anderson, and A. Johnston. Inverse perspective mapping and optic flow: A calibration method and a quantitative analysis. *Image Vision Comput.*, 24(2):153–165, Feb. 2006. ISSN 0262-8856. doi: 10.1016/j.imavis.2005.09.023. URL `http://dx.doi.org/10.1016/j.imavis.2005.09.023`.

S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. The MIT Press, 2005. ISBN 0262201623. URL `https://www.ebook.de/de/product/3701211/sebastian_thrun_wolfram_burgard_dieter_fox_probabilistic_robotics.html`.

S. P. Todd R. Kushner. Progress in road intersection detection for autonomous vehicle navigation, 1987. URL `https://doi.org/10.1117/12.968232`.

S. Tuohy, D. O'Cualain, E. Jones, and M. Glavin. Distance determination for an automobile environment using inverse perspective mapping in opencv. In *IET Irish Signals and Systems Conference (ISSC 2010)*, pages 100–105, June 2010. doi: 10.1049/cp.2010.0495.

R. P. D. Vivacqua, R. F. Vassallo, and F. N. Martins. A low cost sensors approach for accurate vehicle localization and autonomous driving application. In *Sensors*, 2017.

W. Yang, B. Fang, and Y. Y. Tang. Fast and accurate vanishing point detection and its application in inverse perspective mapping of structured road. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(5):755–766, May 2018. ISSN 2168-2216. doi: 10.1109/TSMC.2016.2616490.

Y.-t. Y. Y.-j. C. B. L. L.-s. Z. Yu Cao, Ying Feng. Monocular visual odometry based on inverse perspective mapping, 2011. URL `https://doi.org/10.1117/12.900010`.

J. Zang, W. Zhou, G. Zhang, and Z. Duan. Traffic lane detection using fully convolutional neural network. pages 305–311, 11 2018. doi: 10.23919/APSIPA.2018.8659684.

J. Zhao, B. Liang, and Q. Chen. The key technology toward the self-driving car. *International Journal of Intelligent Unmanned Systems*, 6(1):2–20, 2018. doi: 10.1108/IJIUS-08-2017-0008. URL `https://doi.org/10.1108/IJIUS-08-2017-0008`.