# ALTRUISM: Interpretation of Three-Dimensional Information for Autonomous Vehicle Control

Technical Report TR95SMS3
(This report has now been published in Engineering Applications of Artificial Intelligence [1])

1995

S.M. SMITH
Oxford Centre for Functional Magnetic Resonance Imaging of the Brain (FMRIB),
Department of Clinical Neurology, Oxford University, Oxford, UK
(Previously in Computer Vision and Image Processing Group, DRA Chertsey, DERA, UK)
steve@fmrib.ox.ac.uk
www.fmrib.ox.ac.uk/~steve

**Abstract**

In order to be able to successfully navigate autonomously, a vehicle needs to have an understanding of the local world. It must understand where it is safe to drive and be able to recognize static and moving objects. It must also be able to relate this understanding to the desired route plan, so that progress may be made in the most sensible way. This paper describes an attempt to bridge the large gap between making sparse three dimensional measurements of static scene structure and achieving a useful level of understanding of that structure. In particular, it addresses the need to know where in the scene it is safe to drive.

The system described in this paper is known as ALTRUISM - Automatic Local Three-dimensional Representation Using Image-based Scene Measurements. It is complementary to the ASSET system presented recently in this journal, which detects and tracks moving obstacles from a moving vehicle.

**Key Words**

Autonomous vehicles; obstacle detection; structure-from-motion; egomotion.

## 1 Introduction

This paper presents an attempt to provide useful automatic interpretation of sets of sparse three dimensional world measurements, to allow low level control of autonomous land vehicles (ALVs). The system under development is known as ALTRUISM - Automatic Local Three-dimensional Representation Using Image-based Scene Measurements. Initial results of scene interpretation are promising, and suggest that in the near future direct low level control of an ALV should be possible, even when other simpler cues (e.g., road edges, texture/colour analysis) are not present.

The image processing system which provides the three dimensional scene information for this research is DROID, developed at Roke Manor Research Limited and implemented as a real-time system at DRA Chertsey. DROID provides a list of three dimensional scene measurements, made at well localized places in the image, as the camera is moved through the environment. DROID also produces an estimate of the camera's own motion in the world, given only an initial space scaling factor (enabling the resolution of the scale/speed ambiguity). It works by tracking two dimensional features over time, using a Kalman filter for the three dimensional tracking of each point. For further details on DROID, see [2], [3], [4], [5] and [6].

Previous work in this area at DRA Chertsey has found driveable regions by either looking at local interpolated surface normals and three dimensional height values or by finding a recursive least squares fit of a plane to the three dimensional data output by DROID, to give a two dimensional map of "safe" and "unsafe" regions. However, this work does not go far enough towards providing information that is directly useful to a local route planning system, and also makes no use of temporal integration of information for increased accuracy and stability. The ALTRUISM system attempts to increase the intelligence (and therefore complexity) of the data interpretation stage.

This paper gives details of the work which has been carried out on ALTRUISM so far.

In brief, the following progress has been made:

- A recursive least squares fit of a planar-quadratic "ribbon" to the three dimensional tracked features is computed, ignoring outliers. This gives an estimate of the three dimensional "road" parameters. These four parameters are temporally filtered.
- Those three dimensional features (tracked by DROID) which lie "on" the computed driveable surface are distinguished from those which do not.
- This information is used to compute a radial map of the limits of the driveable surface in the central-tangential plane of that surface. The resulting list of distances may extend further than objects which lie on it but are not perceived as forming its boundary.
- The radial map is filtered within each new estimation to reduce noise and to interpolate missing data, and also filtered temporally.
- The computed planar-quadratic ground surface is used to provide a measure of scale drift within DROID, a serious problem over large sequences unless camera calibration is completely perfect. The estimated height of the camera above the ground, assumed to be constant in reality, is used to drive a feedback loop to force this height measurement to stay constant.
- The scale drift correction allows good estimates of the camera's total journey (in three dimensions) to be made.
- An estimated steering direction command is produced from the driveable region's outline; this can be output to the ALV.
- The whole system has been implemented in real-time and has been run on sequences of several minutes.

Section 2 gives a brief review of previous work on structure-from-motion, egomotion estimation, passive obstacle detection and surface representation. Section 3 presents the research carried out to date on the ALTRUISM system. Results of testing are given in Section 4, and conclusions and future work presented in Section 5.

## 2 Review of past research

Most research relating to the area of image-based three dimensional scene interpretation has to date concentrated on the closely related problems of structure-from-motion (see references [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18] and [19]) and egomotion estimation (see [10], [12], [13], [14], [16], [19], [20], [21], [22], [23], [24] and [25]). However, the method of performing these two tasks is not the subject of this paper, as it is the *interpretation* and not the *estimation* of the information supplied by structure-from-motion and egomotion systems which is of interest.

Some work has been carried out in the past on recognition and modelling of objects using information obtained from structure-from-motion programs (for example, see [26]), but these have tended to be limited to cases where the object/background segmentation problem is solved by hand, or by using simple data, and are not complex enough to cope with the real world situations addressed here.

Of more relevance is research which has specifically attempted to provide warning of static obstacles. In [27] Nelson and Aloimonos use the first order derivatives of the flow field to create suitable functions which give indication of approaching obstacles. The coarse output (which is basically a binary image map of danger or safety) does not easily allow any higher understanding of the world. (See also [28] for theory relating to this work, and [29] where a similar but less well developed idea is used to detect obstacles on the ground plane.) In [30] Enkelmann tests a gradient based "analytical" optical flow estimation method against an edge contour based method and a two dimensional feature point based method, in an experiment to detect a static obstacle viewed from a moving vehicle. Although all three methods appear to detect the obstacle, Enkelmann states a preference for the gradient based method, apparently because of its reliability. However, the DROID system, based on feature points, is designed for exactly this kind of situation, and has reliably found static obstacles in the path of a vehicle. Also, the gradient based method has the weakness, which is mentioned in [30] (and which is shown in the results) that the accuracy of the outline of the obstacle is poor, due to smoothing effects. This problem would be accentuated in the case of moving obstacles. None of the research which attempts to find obstacles from optic flow directly has attempted to provide the higher level more integrated understanding of the world which is needed before intelligent autonomous driving can take place.

In order to begin to achieve some level of understanding of a three dimensional scene, a suitable model or representation must be applied to the data. This provides a reduced amount of data in a useful form; if the representation chosen is an appropriate one, then the amount of *data* will be reduced without the loss of *information*.

Much work has been done on curve and surface descriptions using active contour models (for example, see [31]), B-splines (for example, see [32]), uniform/non-uniform rational/non-rational B-splines (for example, see [33]) and rational Gaussians (see [34]). However, in the case of a surface which can be thought of as being split into driveable and non-driveable parts, the most suitable approach is to fit an appropriate road model to the driveable part, and completely disregard the non-driveable part, as it contains no information of interest to the ALV. The model chosen to fit the road is a planar-quadratic "ribbon". This model achieves a balance between segmentation discrimination ability (and accuracy of fit) and stability. From this representation useful information such as steering direction can be obtained, as shown later.

## 3 Details of research carried out

This section gives details of the work which has been carried out on ALTRUISM so far. The research can be split into two parts; some interesting work has been undertaken to use ALTRUISM to improve internal accuracy and stability of the DROID system over lengthy monocular sequences, but the largest part of the work on ALTRUISM has been the provision of a useful representation of the local scene.

### 3.1 The "road" model

Previous work at DRA Chertsey has used, as input data to calculations of the "driveable region" (or "the road"), local interpolated surface normals, three dimensional height values taken in a fixed global three dimensional co-ordinate frame and positions in the camera's local three dimensional co-ordinate frame. All three of these data types can be computed from the output of DROID. The use of local interpolated surface normals was not very successful due to high noise sensitivity. Because local surface direction depends on the vector differences of three dimensional positions and the other two methods use these positions directly, any noise will naturally become more damaging in the former case. The use of a local three dimensional co-ordinate frame is obviously more robust than a single fixed co-ordinate frame over long image sequences, as there is no susceptibilty to any accumulation of error in the estimated camera position.

The model which is fit to the data must balance the need to accommodate all possible variations within accepted driveable regions with the need to constrain the model to give maximum stability. In other words, the number of degrees of freedom must be not too large and yet not too small. Previous work at DRA Chertsey used a constant height model (one degree of freedom) and a planar fit (three degrees of freedom).

The next more complex isotropic model would naturally be a bi-quadratic fit, with six degrees of freedom. However, transverse curvature and road skew are likely to be very low, and the modelling of these would unnecessarily reduce the ability of the model to distinguish driveable features from non-driveable ones. Hence the model used is a planar-quadratic "ribbon", with four degrees of freedom, namely height, two slope angles, and longitudinal curvature (see figure 1 for qualitative indications of these parameters). In fact, these four degrees of freedom are slightly constrained; both the curvature and the transverse slope are hard limited to give added stability to the model.

The model is fit to the data using a standard least squares fit, with recursive estimation ignoring outliers. The threshold used for the outlier rejection is set at a constant fraction of the estimated distance between the camera and the road. After computation of the fit in the first frame, the starting position for the fit in subsequent frames is taken as the current estimate of the model parameters. Thus convergence is fast and stable.

Once the best estimates for the four model parameters have been found, they are temporally filtered to improve stability and accuracy, using a simple fixed weight filter (of typical update fraction 0.4). The filtered values are used by the other sections of ALTRUISM, and also, as mentioned above, as the first estimate in the least squares fit for the next frame.

The reason for modelling only the driveable region and not the whole observed scene is that far more constraint can reasonably be put on the expected surface types for the driveable region than for "off road" parts of the scene. Also, the non-driveable areas are not of immediate interest to this application, so that as long as the segmentation of driveable and non-driveable points is performed correctly, the non-driveable data can be completely ignored.

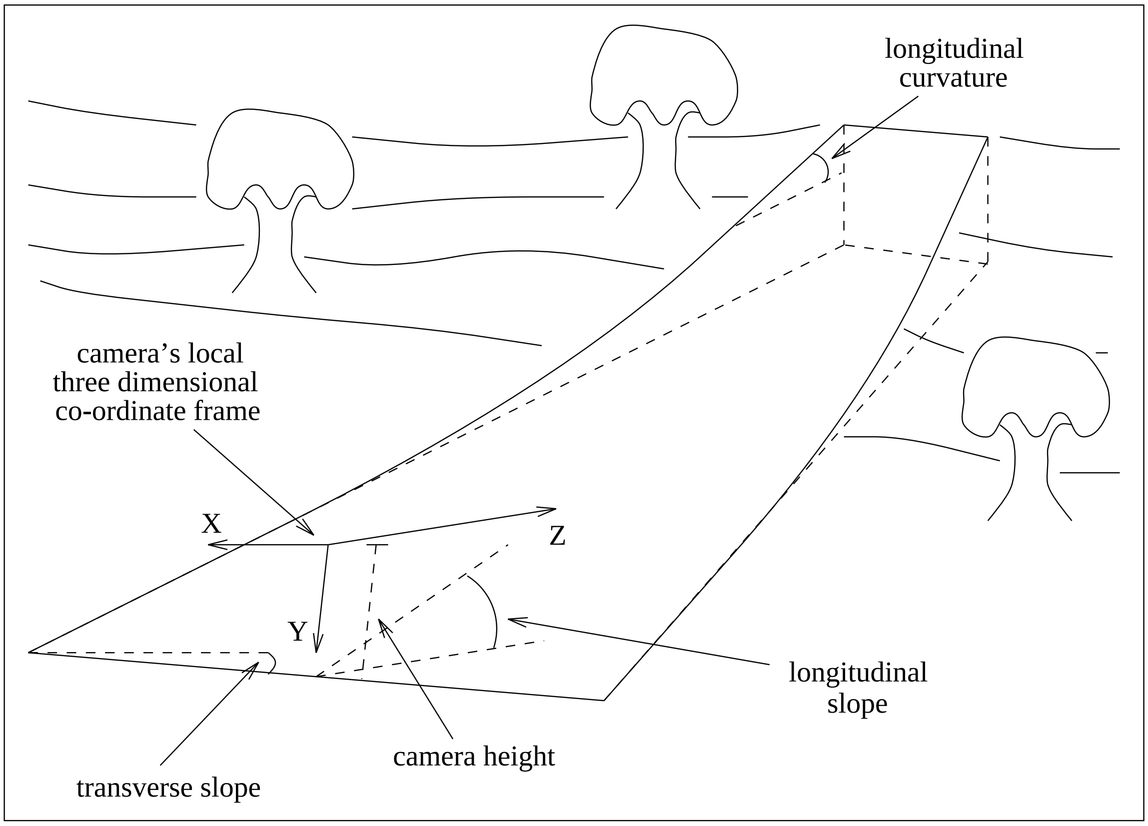


Figure 1: Degrees of freedom in the planar-quadratic model.

### 3.2 Finding the limits of the "road"

As explained above, the process of fitting a model to the three dimensional world measurements provides both the model parameters and the segmentation of scene measurements into "on" and "off" road classes. This information is used to compute a radial map of the limits of the driveable surface in the central-tangential plane of the fitted planar-quadratic model. In other words, the curvature parameter is ignored, and the other three parameters are used to define a plane which provides an approximation of the fitted road model, and into which the three dimensional measurements can be projected so that the road limits may be found.

In this local ground plane estimation a planar radial map is calculated; this is a list of radii found at equal angle increments. Typically, 10 radii fall inside that part of the world which the camera can see. This number gives a good trade-off between high "resolution" and stability due to reasonable numbers of data points falling between consecutive radii.

The resulting list of distances may extend further than objects which lie on the ground plane but are not perceived as forming its boundary. This and other logical expectations about the radial map are implemented by using the following stages in its estimation:

1. Between each pair of radii, the nearest "off road" point to the origin (the position on the ground plane underneath the camera) and the furthest "on road" point are calculated.

2. These two measurements are combined to give one radial length – it is set to the greater of the two measurements. This method is chosen to give the best results in the absence of dense data at the boundary, a common occurrence. It will also allow the separate modelling of static obstacles within the driveable region in future.
3. Next a three point median filter is applied to the estimated radii, ignoring radii for which no reliable measurements have been made. This provides spatial smoothing.
4. The "missing data" where no reliable measurements have been made are filled in where appropriate, using linear interpolation between reliable radii. Measurements are deemed "reliable" or not depending on the number of feature points which have contributed to their estimation.
5. The radial map thus produced is combined with a temporally filtered radial map to give a new filtered estimate, using a simple linear combination. This finally gives the best estimate of the current driveable limits in the ground plane.

### 3.3 DROID scale drift correction

The computed planar-quadratic ground surface is used to provide a measure of scale drift within DROID, a serious problem over long sequences unless camera calibration is completely perfect, which is virtually impossible to achieve in practice. As the only input relating image motion to three dimensional world co-ordinates is the single initial camera displacement,[1] it is not surprising that over many frames DROID's absolute scale gradually drifts. This drift can be countered by using external inputs to DROID, such as vehicle odometry readings. This, however, may not always be available, and a self-correcting system is preferable. It is also recognized that with tasks such as past route estimation, the more independent sources of information that can be combined, the better the final estimation will be.

The first stage in the drift correction is the estimation of the height of the camera above the computed ground surface (i.e., parameter $c$ in the surface fit). A heavily filtered version of this height is calculated during the first 50 frames after reliable surface fitting has begun. This height is then assumed to be the exact height (or, more correctly, the actual height is "defined" to be this height), and future height measurements are compared against this. In all future frames, the comparison of estimated camera height with "real" height is used to produce a scaling factor which is fed into DROID. DROID uses the previous frame's calculated egomotion as the initial estimate for the present frame's calculations, so this is a good point to feed in the scaling correction. The corrections are very close to 1 (either 0.97 or 1.03, depending on the sign of the difference between the estimated camera height and the "real" height) so that stability is not endangered, and thus avoiding the need for in-depth understanding of the feedback dynamics.

[1] This is often known as resolution of the scale-speed ambiguity.

### 3.4 Production of a steering command

The filtered radial map of the driveable region can be used to create a steering command or estimated future road direction. The longest parts of the radial map can be expected to give an indication of the direction of the road in front of the camera. To obtain a quantitative measure of road direction, the following calculation is made:

$$direction = \frac{\sum \theta_i r_i^4}{\sum r_i^4}, \tag{1}$$

where $r_i$ is the $i$th radius and $\theta_i$ is the angle associated with it. The fourth power is used to give extra weight to the larger radii. It is obviously important that straight ahead be given an angle of zero, and that negative values be used to one side of this, for this formulation to work.

The steering angle is temporally filtered for stability.

### 3.5 Real-time implementation

The DROID-ALTRUISM system has been implemented at DRA Chertsey to run live on the ROVA (ROad Vehicle Autonomous) i860-based image processing system; see figures 2, 3 and 4.

Images are digitized and stored until read using Datacube equipment. Feature detection is performed in custom hardware supplied by Roke Manor Research Limited, which provides a list of features in VME address space at full frame rate (25 Hz), being fed images by the Datacube Maxbus digital image bus. This list is read by a Parsytec VME/Transputer link interface, which sends the feature lists down a Transputer link to the first Transtech Transputer/i860 board, straight into i860 memory.

DROID runs on this first i860 without any need for parallelization. It runs at an average speed of 23 Hz. Because this is less than 25 Hz, and, more importantly, because the standard deviation is quite high (about 3 Hz), DROID is run at half frame rate. The knowledge that this module will always complete in the expected time makes system design much simpler. The list of three dimensional world measurements and the camera's estimated egomotion are sent down a Transputer link to the second Transtech board.

ALTRUISM runs on the second i860. The current system would easily run at full frame rate, but runs at half frame rate here to match the input data rate. The results of processing can be converted to video display information; the three dimensional features can be shown colour coded according to whether they are "on" or "off" road, the radial map representing the driveable region can be shown, either as plan view, or re-projected back into the world and then back into the image, and, finally, the steering angle can be represented graphically. All of this video overlay information is sent down a Transputer link to the original Parsytec interface board. This information is sent at full frame rate, even though it is only changed on every other frame, so that output graphics are smooth and continuous.

The Parsytec board performs graphical overlay by over-writing the latest image in the Datacube framestore. Images are not stored for longer than one frame period in this double-buffered framestore, so that graphics are not overlaid on exactly the right image, but the incorrect positioning is barely visible by eye.

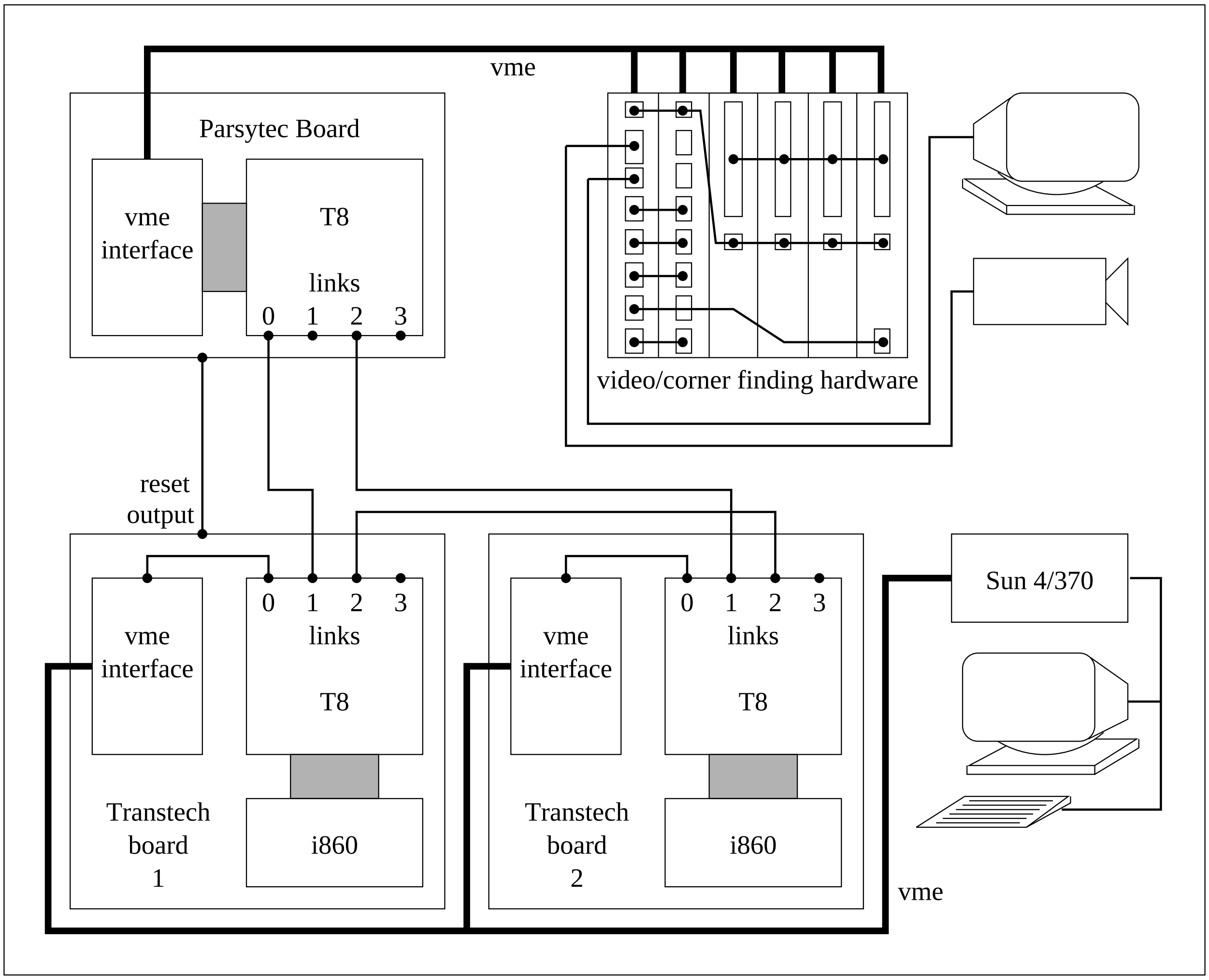

Figure 2: The ROVA i860-based image processing system with hardware front end.

## 4 Results

In this section example outputs from ALTRUISM are presented.

The sequence used to produce the examples in this section was taken on a seven minute circuit around a rough track at DRA Chertsey. (A map of this track forms part of figure 10.) As explained earlier, the DROID-ALTRUISM system ran at half frame rate. For every fifth processed frame, details of the estimated camera position and of the road model parameters were output to file. This data was used to create the graphs shown below.

The four road model parameters are first plotted, where the model used is:

$$Y = aX + bZ + dZ^2 + c, \tag{2}$$

where the $Y$ axis points straight down, the $X$ axis points from right to left and the $Z$ axis points away from the camera in the direction of the optic axis (see figure 1).

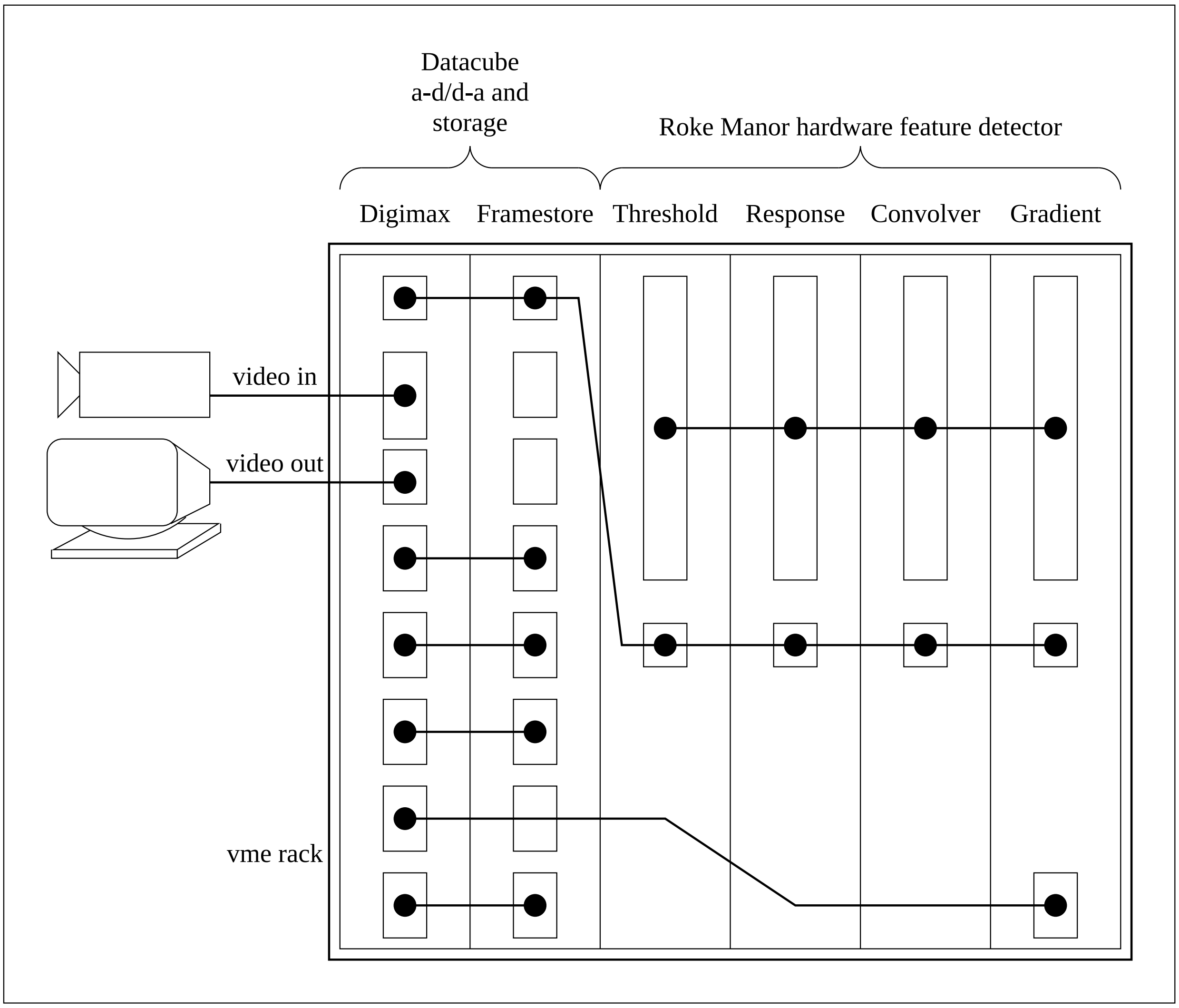


Figure 3: Connections and setup for the hardware feature detector.

In figure 5, the filtered parameter $a$ is plotted against processed frame number. This is a measure of transverse slope.

In figure 6, the filtered parameter $b$ is plotted against processed frame number. This is a measure of longitudinal slope.

In figure 7, the filtered parameter $d$ is plotted against processed frame number. This is a measure of longitudinal curvature.

In figure 8, the filtered parameter $c$ is plotted against processed frame number. This is the height (in metres) of the camera above the road. Also shown on this graph is the "very" filtered height estimate (obtained by using a lower update fraction) which is used by the scale drift correction stage. The dashed line shows the "very" filtered camera height when no scale drift correction is used. It is clear that scale drift is a serious problem without the correction stage in ALTRUISM; see also figure 11.

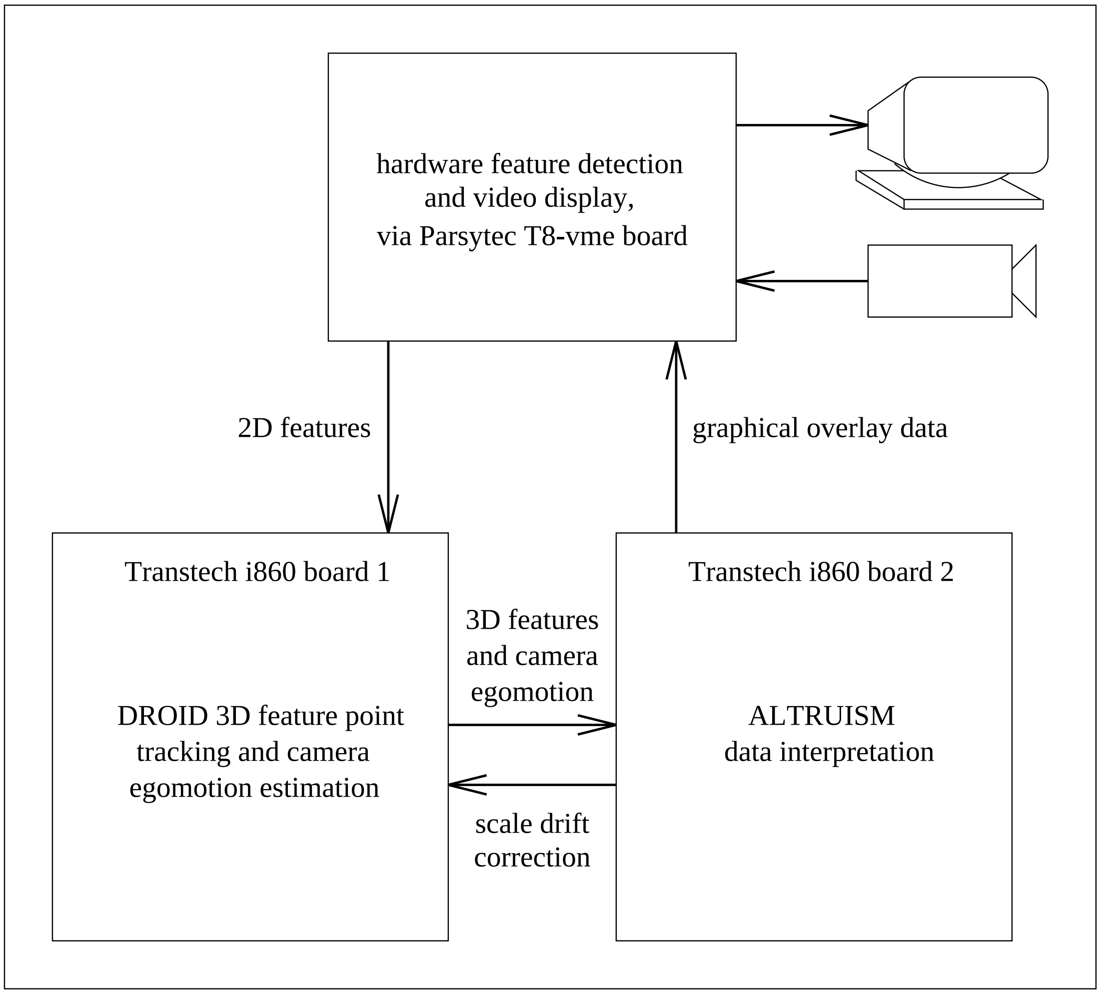


Figure 4: Overview of the live DROID-ALTRUISM setup.

With respect to the accuracy of these parameter estimates; for each parameter, the error can be thought of as split into two parts; high frequency and low frequency error. The highest frequency variations (visible in and measureable from the figures) in the parameters give an indication of the level of accuracy of the instantaneous surface fits; this will be due to inaccuracies in feature point segmentation and the least-squares fit process. The low frequency errors are due to inaccuracies in the 3D data output from DROID; it would be very difficult to estimate these errors, but it can be assumed, given the quality of the highest level results, that these errors are not significant.

The way in which errors in the road surface parameters affect the accuracy of the estimated vehicle route is indirect, and it seems likely that only a mistake in the ALTRUISM algorithms or programming (causing systematic errors in these parameters) would result in propagation of errors to the route estimation, even though this estimation is dependent on the scale drift correction, itself driven by the parameter estimation; this is due to the effective integration of errors in parameter estimation in this part of the data flow process.

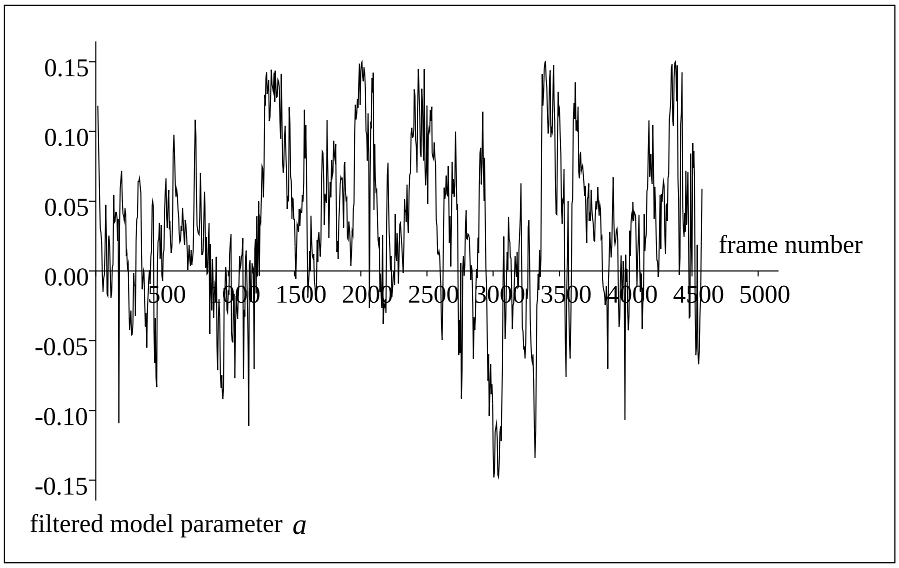


Figure 5: A plot of filtered model parameter $a$ against frame number.

In figure 9 the $Y$ co-ordinate of the estimated camera position (i.e., absolute camera height, in the initially established global three dimensional co-ordinate frame) is plotted against processed frame number. Unfortunately there are no world measurements to compare this data with. The final offset from zero is due to the slight incompleteness of the circuit (see figure 10).

In figure 10 the $X$ co-ordinate of the estimated camera position is plotted against $Z$ (i.e., this is a plan view of the camera's trajectory, in the initially established global three dimensional ground plane). This trajectory is shown as a thick dashed line, superimposed upon a line drawing map of the area, produced photographically for the site.

Figure 11 compares the estimated camera trajectory shown in figure 10 with that estimated when the scale drift correction stage of ALTRUISM is not used. The features of the track can still be made out, but the scale drift problems are obvious. This kind of discrepancy is regularly found when extended sequences are processed.

Figure 12 shows an example of the feature segmentation stage of ALTRUISM, on one of the images in the rough track sequence. The model fitting has provided information about which three dimensional features are "on road" (dark features with light borders) and which are "off road" (light features with dark borders). The discrimination is working well.

Figure 13 shows an example of the driveable region radial map estimation stage of ALTRUISM. The map is projected back into the image.

Figure 14 shows the same radial map from plan view, as well as steering angle estimation. The bottom vertex of the radial map corresponds to the camera's position, and the angle subtended by the the first non-zero radial estimates corresponds to the field of view. The vertical line indicates the local $Z$ axis, i.e., "straight ahead". The non-vertical line indicates the predicted necessary steering direction (in the plan view frame).

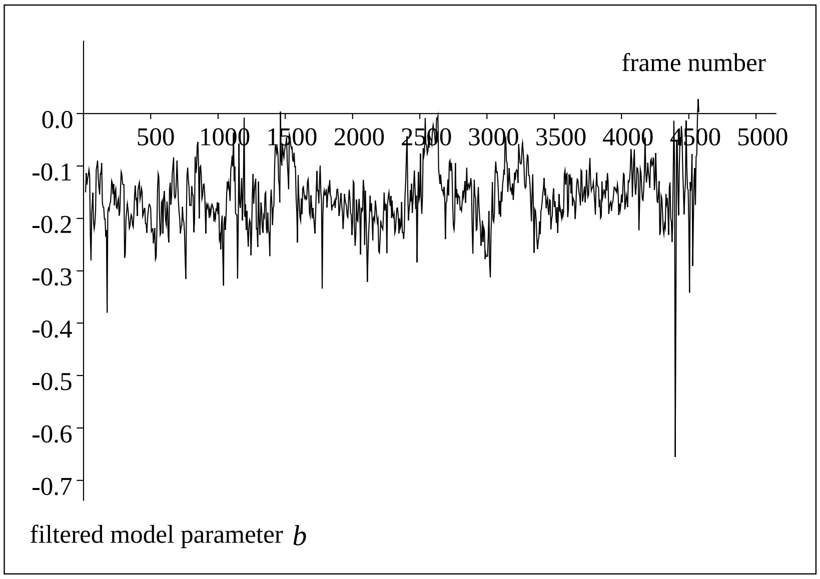


Figure 6: A plot of filtered model parameter $b$ against frame number.

# 5 Conclusions and future work

This paper has given details of initial development and testing of ALTRUISM, Automatic Local Three-dimensional Representation Using Image-based Scene Measurements. ALTRUISM takes a list of three dimensional world measurements and estimated camera movement and produces a higher level understanding of the relevant parts of the world. The measurements (also known as features) are segmented into two groups, "on road" and "off road", at the same time as a four parameter linear-quadratic fit is made to the "on road" data. The "on road" driveable region's shape is modelled by a temporally filtered radial map estimated in the central-tangential plane of the region. This shape is used to estimate a required steering angle prediction. ALTRUISM also provides a method for scale drift self-correction, based on the assumption of constant camera height above the ground. This correction is fed back into the DROID system which is then enabled to work indefinitely without scale drift.

ALTRUISM has been implemented in real-time on the ROVA i860-based image processing system, with DROID also running in real-time on this system. Very promising initial results have been obtained, some of which were presented in this paper.

Future work will include:

1. Modelling and tracking static obstacles which lie within the driveable region.

2. Testing ALTRUISM on ROVA, the ALV at DRA Chertsey, under closed-loop control.

3. Integrating the DROID-ALTRUISM system with the ASSET system (see [35]). ASSET will provide pre-segmentation of moving obstacles in the image, to enable the DROID-ALTRUISM system to continue functioning in the presence of moving obstacles.

4. Integration of the above with other scene analysis methods, such as road edge detection

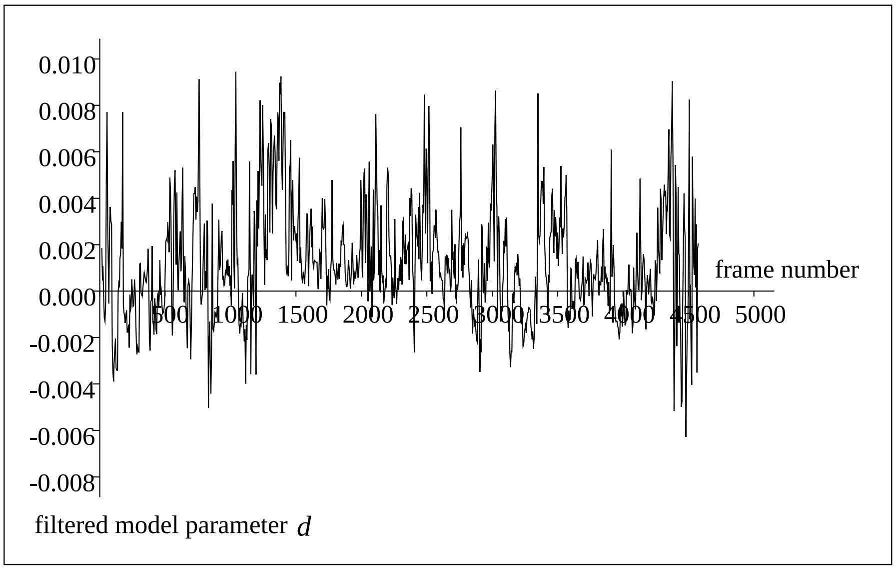


Figure 7: A plot of filtered model parameter $d$ against frame number.

and texture/colour segmentation, to provide greater robustness of driveable region estimation.

These areas will be addressed in future phases of this project.

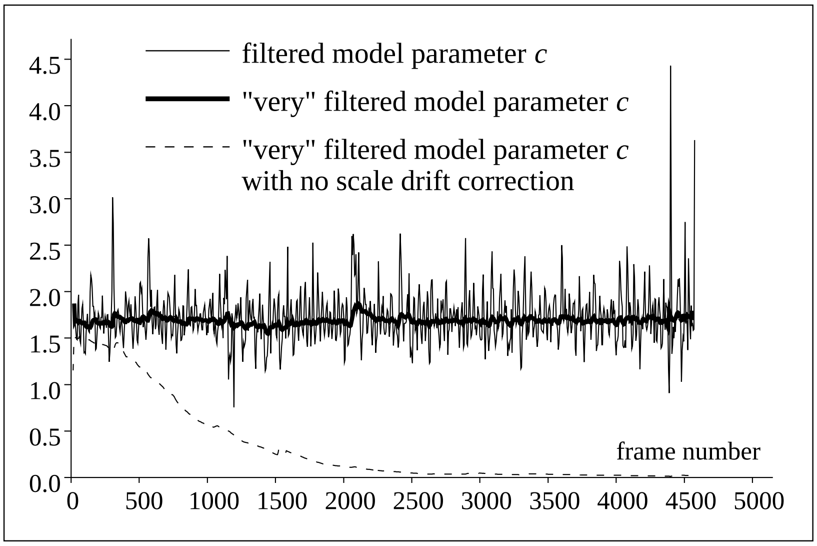


Figure 8: A plot of filtered model parameter $c$, "very" filtered model parameter $c$ and "very" filtered model parameter $c$ when no scale drift correction is used, against frame number.

## References

[1] S.M. Smith. ALTRUISM: Interpretation of three-dimensional information for autonomous vehicle control. *Engineering Applications of Artificial Intelligence*, 8(3):271–280, 1995.

[2] C.G. Harris. Determination of ego-motion from matched points. In *3rd Alvey Vision Conference*, pages 189–192, September 1987.

[3] C.G. Harris and J.M. Pike. 3D positional integration from image sequences. *Image and Vision Computing*, 6(2):87–90, 1988.

[4] C.G. Harris and M. Stephens. A combined corner and edge detector. In *4th Alvey Vision Conference*, pages 147–151, 1988.

[5] D. Charnley and R.J. Blissett. Surface reconstruction from outdoor image sequences. *Image and Vision Computing*, 7(1):10–16, 1989.

[6] S.M. Smith. *Feature Based Image Sequence Understanding*. D.Phil. thesis, Robotics Research Group, Department of Engineering Science, Oxford University, 1992.

[7] A. Blake and R. Cipolla. Robust estimation of surface curvature from deformation of apparent contours. Technical Report OUEL 1797/89, University of Oxford, 1989.

[8] R.C. Bolles, H.H. Baker, and D.H. Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *Int. Journal of Computer Vision*, 1:7–55, 1987.

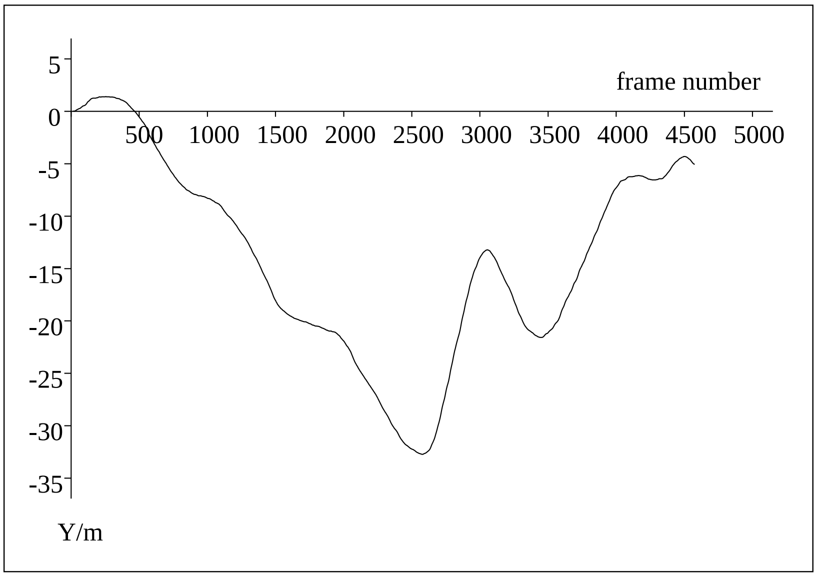


Figure 9: A plot of absolute (global) camera height against frame number.

[9] R. Cipolla, Y. Okamoto, and Y. Kuno. Robust structure from motion using motion parallax. In *Proc. 4th Int. Conf. on Computer Vision*, pages 374–382, 1993.

[10] R. Deriche and O. Faugeras. Tracking line segments. In O. Faugeras, editor, *Proc. 1st European Conf. on Computer Vision*, pages 259–268. Springer-Verlag, 1990.

[11] O.D. Faugeras. On the motion of 3D curves and its relationship to optical flow. In *Proc. 1st European Conf. on Computer Vision*, pages 107–117, 1990.

[12] O.D. Faugeras, F. Lustman, and G. Toscani. Motion and structure from motion from point and line matches. In *Proc. 1st Int. Conf. on Computer Vision*, pages 25–34, 1987.

[13] O.D. Faugeras and T. Papadopoulo. A theory of the motion fields of curves. *Int. Journal of Computer Vision*, 10(2):125–156, 1993.

[14] X. Hu and N. Ahuja. Motion and structure estimation using long sequence motion models. *Image and Vision Computing*, 11(9):549–569, 1993.

[15] C.P. Jerian and R. Jain. Structure from motion – a critical analysis of methods. *IEEE Trans. on Systems, Man and Cybernetics*, 21(3):572–588, 1991.

[16] D.T. Lawton. Processing translational motion sequences. *Computer Vision, Graphics and Image Processing*, 22:116–144, 1983.

[17] D.W. Murray. Algebraic polyhedral constraints and 3D structure from motion. In *Proc. 5th Alvey Vision Conference*, pages 215–220, 1989.

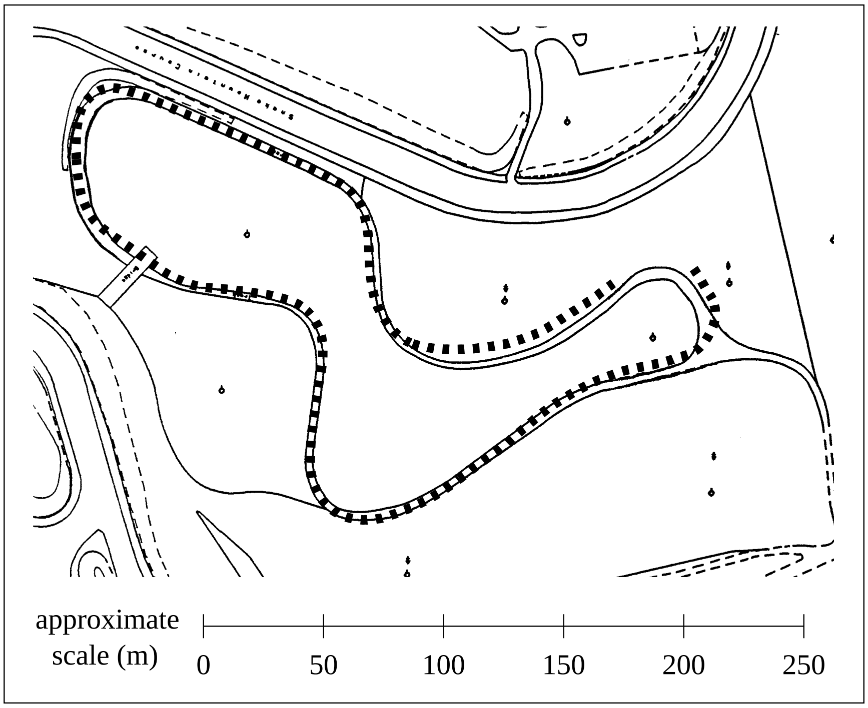


Figure 10: Estimated route taken by the camera with ALTRUISM providing scale drift correction, overlaid on a map of the area.

[18] G. Sandini and M. Tistarelli. Active tracking strategy for monocular depth inference over multiple frames. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(1), January 1990.

[19] C. Tomasi and T. Kanade. Shape and motion without depth. In *Proc. Image Understanding Workshop*, 1990.

[20] P. Beardsley, D. Sinclair, and A. Zisserman. Ego-motion from six points. Technical report, University of Oxford, 1991.

[21] W. Burger and B. Bhanu. Estimating 3-D egomotion from perspective image sequences. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(11):1040–1058, 1990.

[22] O.D. Faugeras and S. Maybank. Motion from point matches: Multiplicity of solutions. *Int. Journal of Computer Vision*, 4:225–246, 1990.

[23] D.J. Heeger and A. Jepson. Visual perception of three dimensional motion. *Neural Computation*, 2:129–137, 1990.

[24] B.K.P. Horn and E.J. Weldon. Direct methods for recovering motion. *Int. Journal of Computer Vision*, 2:51–76, 1988.

[25] S. Negahdaripour and S. Lee. Motion recovery from image sequences using only first order optical flow information. *Int. Journal of Computer Vision*, 9(3):163–184, 1992.

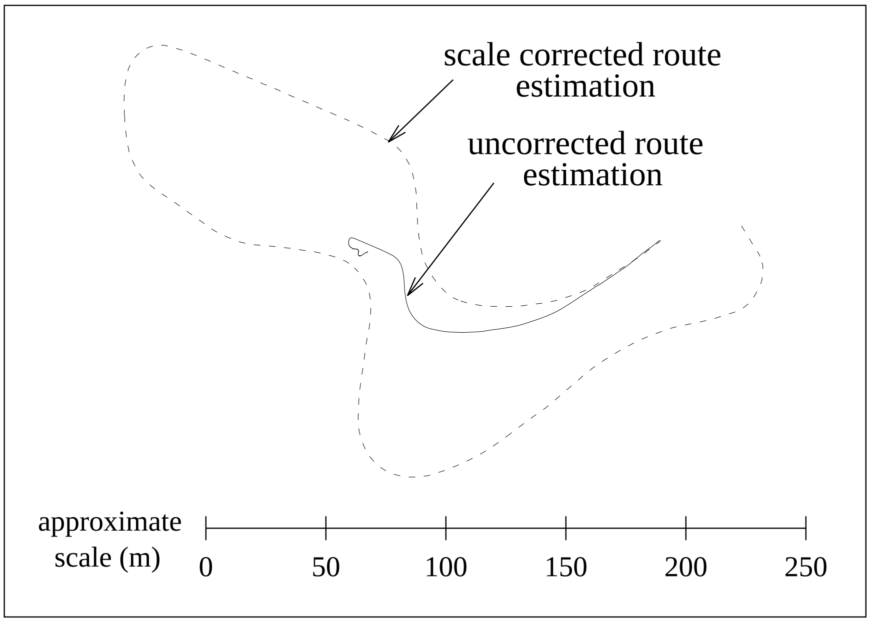


Figure 11: Estimated route taken by the camera with and without scale drift correction.

[26] D.W. Murray, D.A. Castelow, and B.F. Buxton. From an image sequence to a recognized polyhedral object. In *Proc. 3rd Alvey Vision Conference*, pages 201–210, 1987.

[27] R.C. Nelson and J. Aloimonos. Using flow field divergence for obstacle avoidance: towards qualitative vision. In *Proc. 2nd Int. Conf. on Computer Vision*, pages 188–196, 1988.

[28] M. Subbarao. Bounds on time-to-collision and rotational component from first-order derivatives of image flow. *Computer Vision, Graphics and Image Processing*, 50:329–341, 1990.

[29] G.-S. Young, T.-H. Hong, M Herman, and J.C.S. Yang. New visual invariants for obstacle detection using optical flow induced from general motion. In *IEEE Workshop on Applications of Computer Vision*, pages 100–109, 1992.

[30] W. Enkelmann. Obstacle detection by evaluation of optical flow fields from image sequences. In *Proc. 1st European Conf. on Computer Vision*, pages 134–138, 1990.

[31] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. In *Proc. 1st Int. Conf. on Computer Vision*, pages 259–268, 1987.

[32] A. Blake, R. Curwen, and A. Zisserman. A framework for spatiotemporal control in the tracking of visual contours. *Int. Journal of Computer Vision*, 11(2):127–145, 1993.

[33] P.J. Besl. The free-form surface matching problem. In H. Freeman, editor, *Machine Vision for Three-Dimensional Scenes*. Academic Press, 1990.

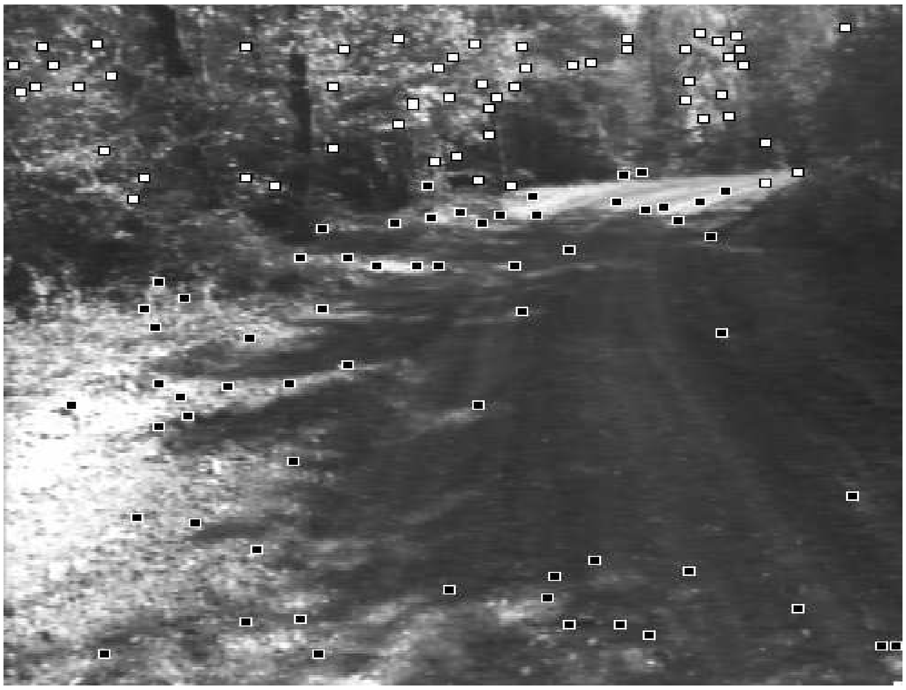

Figure 12: Tracked three dimensional features marked as "on" and "off" road.


[34] A. Goshtasby. Design and recovery of 2-D and 3-D shapes using rational Gaussian curves and surfaces. *Int. Journal of Computer Vision*, 10(3):233–256, 1993.

[35] S.M. Smith and J.M. Brady. A scene segmenter; visual tracking of moving vehicles. *Engineering Applications of Artificial Intelligence*, 7(2):191–204, April 1994.

Figure 13: Estimated driveable region outline.

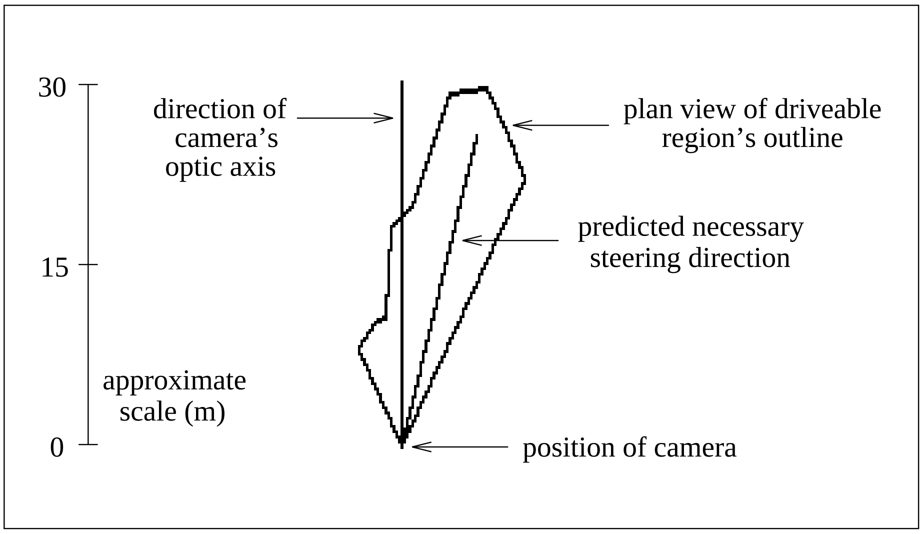


Figure 14: Plan view of the driveable region's outline, together with predicted necessary steering direction.