Visual Vehicle Egomotion Estimation using the Fourier-Mellin Transform

Roland Goecke[1,2], Akshay Asthana[3], Niklas Pettersson[1,2] and Lars Petersson[1,2]
[1]Vision Science, Technology and Applications, NICTA Canberra Research Laboratory, Canberra, Australia
[2]Dept. of Information Engineering, RSISE, Australian National University, Canberra, Australia
[3]Computer Science & Information Technology Dept., Jaypee Institute of Information Technology, Noida, India
roland.goecke@anu.edu.au akshay.asthana@gmail.com
{niklas.pettersson, lars.petersson}@nicta.com.au

***Abstract*— This paper is concerned with the problem of estimating the motion of a single camera from a sequence of images, with an application scenario of vehicle egomotion estimation. Egomotion estimation has been an active area of research for many years and various solutions to the problem have been proposed. Many methods rely on optical flow or local image features to establish the spatial relationship between two images. A new method of egomotion estimation is presented which makes use of the Fourier-Mellin Transform for registering images in a video sequence, from which the rotation and translation of the camera motion can be estimated. The Fourier-Mellin Transform provides an accurate and efficient way of computing the camera motion parameters. It is a global method that takes the contributions from all pixels into account. The performance of the proposed approach is compared to two variants of optical flow methods and results are presented for a real-world video sequence taken from a moving vehicle.**

I. INTRODUCTION

The problem of estimating the egomotion of a camera has been an active area of research for many years. It has applications in many computer vision and robotics areas such as scene reconstruction by structure-from-motion, autonomous navigation, and obstacle detection and avoidance. In the work presented here, we are interested in estimating the egomotion of a monocular camera system installed in a vehicle and thus in estimating the egomotion of the vehicle, which is a key requirement for vision-based driver assistance, collision avoidance and autonomous driving. It is one of the first, yet very important steps in detecting and tracking independently moving objects (e.g. other vehicles, pedestrians, cyclists) when the observer is also moving.

Various sensors have been proposed to use in such a motion estimation scenario, for example, monocular and stereo camera systems, GPS-based systems, laser or radar systems. The latter two are generally considered to be more reliable in 'bad' (in terms of illumination) conditions because they do not rely on the presence of light. However, these systems are also significantly more expensive than a camera system and with the widespread availability of good quality cameras at low prices, computer vision-based systems are particularly appealing. While it is possible to use GPS-based systems for egomotion estimation, there are reliability issues with such systems when there is no direct line of sight to one of the satellites, for example, in a built-up urban environment ('canyon effect') or in tunnels. Vision-based systems do not suffer from these drawbacks, but their performance can decrease in bad weather and low light conditions. However, with the availability of camera technology beyond the visible spectrum, e.g. far infrared cameras that measure emitted heat, many of the obstacles can be overcome.

In the work presented here, we also follow a vision-based approach. Many vision-based approaches have been proposed in the literature, e.g. [1], [2], [3], [4], [5], [6], [7]. Section II provides an overview of some of these approaches. These include both monocular and stereo camera systems. The latter, if calibrated, allow the recovery of 3D world coordinates for objects in the scene, in particular depth, so that it is easily possible to derive the camera's motion parameters and hence the velocity, orientation and direction of the vehicle. On the other hand, monocular camera system, even when calibrated, cannot recover depth directly. Nevertheless, it is possible to derive the incremental motion of the camera by mapping camera coordinates to corresponding points on the ground plane and determining the displacement of tracked features [8]. At the core of these approaches is the image registration problem, i.e. how are pairs of images in a sequence spatially related to each other?

There are many image registration methods in the literature [9]. Common methods include optical flow methods, e.g. [10], [11], and methods that find a sufficient number of point correspondences by finding stable local image features, e.g. SIFT [12], Harris corners [13], and maximally stable extremal regions [14]. In this paper, we propose to solve the image registration problem by using the Fourier-Mellin Transform (FMT) which can recover the rotation, translation and scale parameters of the transformation required to register one image to another one [15], [16]. The FMT is a global method that takes the contributions from all pixels into account. It is an efficient and accurate method for pairs of images where the distortions due to the perspective projection in the camera are not too large, such as in pairs of consecutive video frames, so that the assumption of a rigid transformation between the two images holds.

The remainder of this paper is organised as follows. In Section II, an overview of related work is given. Section III presents the theory of the FMT which is at the centre

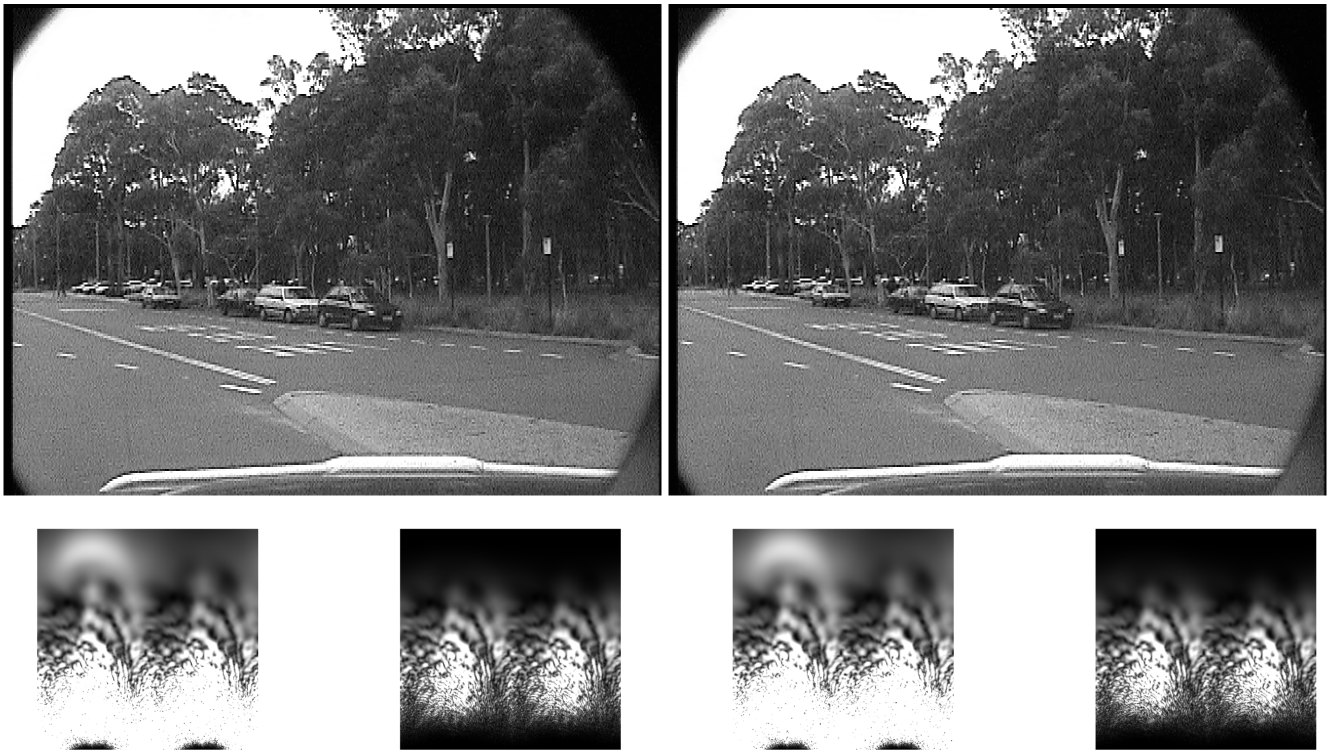

Fig. 1. An example of two consecutive images in the sequence, exhibiting a left turn by the vehicle, and their respective frequency log-polar and windowed frequency log-polar plots, as used in the FMT approach.

of the proposed egomotion estimation method. Section IV then discusses approaches taken to estimate a vehicle's egomotion, including the method proposed in this paper. The results of the experimental evaluation of the proposed method are shown in Section V and compared to existing methods. Finally, Section VI concludes the paper with a summary and an outlook to future work.

II. Related Work

A. Egomotion Estimation

Various methods to compute the egomotion of a mobile camera system have been proposed. A brief overview is given in the following.

A stereo camera system was used in [2]. 3D points are calculated from the calibrated stereo camera system and then optical flow is used to establish correspondences between points. The 6D motion parameters (rotation and translation) are then computed using a least-squares closed form solution rotation quaternions. A smoothness motion constraint is applied to reject inconsistent motions.

Egomotion estimation using a monocular system was presented in [3]. Rather than treating the problem as a pure computer vision problem, the authors incorporate knowledge about the motion behaviour of objects, as set out by the laws of Physics. In addition, a 3D road model with both horizontal and vertical curvature as well as a dynamic model of the egomotion are used. A Kalman filter is then employed to estimate the models parameters.

Egomotion estimation based on optical flow is described in [7]. Optical flow is computed using the Lucas-Kanade method [11]. As computing the optical flow over an entire image is computationally expensive, the authors concentrate on ways to constrain the number of flow computations to a fixed number. Optical flow is only computed at a limited number of sample points which are chosen by Monte Carlo sampling and a vector of random variables which are distributed according to an initialisation distribution function. Point correspondences are then determined using an iterative linearised method which performs better than the 8-point algorithm and the Levenberg-Marquardt algorithm (see [17] for details on these).

Another approach using optical flow is presented in [6]. Here, the authors apply the fact that optical flow vectors generated by a planar surface, e.g. the road surface, conform to a specific equation whose parameters are determined by the relative motion and orientation between camera and surface. In doing so, it is possible to calculate the motion parameters in 3D world coordinates. An Extended Kalman Filter is employed to compute the nonlinear relationship between the optical flow vectors and the motion parameters.

Finally, [4], [5] propose a direct method where each pixel contributes to the measurements, as a way of overcoming the problem of unreliable feature points in cluttered scenes with independently moving objects. The direct approach has the advantage that it avoids the computation of optical flow and feature tracks. Instead, a global probability function is used to combine the measurements, so that the motion parameters can be determined.

In the work presented here, we follow an approach similar to [8]. However, instead of using optical flow, we make use of the FMT to register the images. Therefore, we will first describe the FMT in Section III, before describing the method in detail in Section IV.

B. Image Registration

Image registration describes the process of aligning or matching two or more images in such a way that the objects and features shown in the images overlap in such a fashion that the borders of the images are not visible. A common assumption is that one deals with the case of rigid transformation, i.e. only rotation (θ), translation (t_x, t_y) and scale (s_x, s_y) occur. The transformation between two images $I_1(x, y)$ and $I_2(u, v)$ can then be described by:

$$x = s_x\, u\, cos(\theta) - s_x\, v\, sin(\theta) + t_x \quad (1)$$

$$y = s_y\, u\, sin(\theta) + s_y\, v\, cos(\theta) + t_y \quad . \quad (2)$$

However, this assumption does not hold for a moving camera and perspective projections where the distortions can no longer be described by a rigid transformation, but instead require an affine transformation. Nevertheless, in practice, the assumption of a rigid transformation approximately holds for images taken at short intervals which is the case here. The effects due to perspective projections can then be neglected.

Image registration methods can be categorised based on the algorithms used: correlation methods, point mapping, Fourier methods, and elastic model-based matching. For a review, see, for example, [9]. Correlation methods often find their application in optical flow computation, where small parts of the image are individually tracked from image to image in a sequence, thus creating a mapping between the images. Point mapping methods attempt to find correspondences in salient image features. In a first step, local image features such as SIFT [12], Harris corners [13] or MSERs [14] are computed, before an optimisation step finds the best transformation parameters that correspond to the matched local image features. However, feature points can be unreliable in cluttered scenes. Fourier methods take advantage of the properties of Fourier space [15], [16]. This is also the approach taken in our work and we will present the theoretical background of the Fourier methods in Section III. In recent years, non-rigid image matching methods have become popular in some areas of computer vision, e.g. for face tracking [18], [19], but these are not considered here because vehicles, urban environments and landscapes can be seen as largely rigid objects.

III. THE FOURIER-MELLIN TRANSFORM

Fourier space image registration methods provide a way to recover all rigid transformation parameters, i.e. rotation, translation and scale. They differ from other registration methods in that they search for the optimal match in the frequency domain. These methods make use of the Fourier Shift Theorem and the Fourier Rotation Theorem to provide invariance to rotation, translation and scale. Image registration is then performed by phase correlation of the cross-power spectra [15], [16], [20]. It is possible to compute the Fourier Transform of an image efficiently by using the Fast Fourier Transform (FFT).

Let $F_1(\xi, \eta)$ and $F_2(\xi, \eta)$ be the Fourier transforms corresponding to images $I_1(u, v)$ and $I_2(u, v)$. If I_1 and I_2 are related by a rotation θ and translation (u_0, v_0), both in the image plane, their Fourier transforms are related by

$$\begin{aligned} F_2(\xi, \eta) &= e^{-j2\pi(\xi u_0 + \eta v_0)} \times \\ & \quad F_1(\xi cos(\theta) + \eta sin(\theta), -\xi sin(\theta) + \eta cos(\theta)) \end{aligned} \quad (3)$$

Let M_1 and M_2 be the magnitudes of F_1 and F_2, respectively, which are related by

$$M_2(\xi, \eta) = M_1(\xi cos(\theta) + \eta sin(\theta), -\xi sin(\theta) + \eta cos(\theta)) \ . \quad (4)$$

To recover both rotation and scale simultaneously, the Fourier magnitude spectra are transformed to a log-polar representation (ρ, γ). M_1 and M_2 are then related by

$$M_2(\rho, \gamma) = M_1(\rho/s, \gamma - \theta) \quad (5)$$

where s is the scaling factor, $\xi = log(\rho)$ and $\eta = log(s)$.

The cross-power spectrum is then defined as

$$\frac{F_1(\xi, \eta) F_2^{'*}(\xi, \eta)}{|F_1(\xi, \eta) F_2^{'}(\xi, \eta)|} = e^{-j2\pi(\xi u_0 + \eta v_0)} \quad (6)$$

where F^* is the complex conjugate of F. The Fourier Shift Theorem guarantees that the phase of the cross-power spectrum is equivalent to the phase difference between the images. Then, by taking the inverse Fourier transform, a function can be obtained that is approximately zero everywhere except at the optimal registration point. This phase correlation technique is first applied to recover the scale s and the rotation angle θ, before the translation is found with another phase correlation on the scaled and rotated image. Once the images have been registered, the camera motion parameters can be easily recovered using the egomotion estimation method described in the next section, without having to explicitly compute optical flow vectors.

Figure 1 shows an example of two consecutive images in the sequence, together with their frequency log-polar and windowed frequency log-polar plots, respectively. The vehicle is performing a slight left turn at that point in time.

IV. EGOMOTION ESTIMATION

In this section, the egomotion estimation algorithm is described in detail. The approach is akin to Campbell *et al.* [8], but avoids explicit computation of optical flow vectors. Instead, the FMT is used to provide a global image registration result for image pairs. In the following, first, an overview of how a mapping between the camera coordinates and points on the ground plane is computed is given (Sec. IV-A), before methods dealing with outliers in the computed translation parameters are presented (Sec. IV-B).

Over a short period of time, e.g. the time difference Δt between two consecutive images I_{t-1}, I_t in the sequence, the vehicle's motion on the ground plane can be decomposed into a rotation and a translation component, which are estimated separately from different parts of the images in [8]. Distant points will exhibit only small amounts of translation induced by the parallax, while any amount of rotation will dominate the apparent movement of such points. In contrast, the rotation will cause both nearby and distant points to move by the same angle. In [8], each optical flow vector above the horizon (= distant points) is back-projected onto a vertical cylindrical coordinate system centred on the focal point of the camera and the rotation angle estimate is taken as the median of the observed angular displacements. In doing so, effects caused by a change in rotation angle can be removed from the optical flow field and any remaining effects are due to translation. The pure translation parameters in the ground plane that best fit with the observed optical flow are then found. In [8], the optical flow vectors are back-projected onto the ground plane. For a calibrated camera, the length of each back-projected optical flow vector then corresponds to the actual displacement on the ground plane. To overcome tracking errors caused by perturbations, a robust estimate of the translation component is derived from the median of the y-displacements.

In the FMT approach, the point correspondences for all image pixels are known because of the global nature of image registration by FMT. Thus, we avoid having to compute large numbers of optical flow vectors and to deal with noisy or erroneous optical flow estimates. In contrast, the camera's rotation and translation parameters can be directly computed from the image registration parameters s, θ, and (u_0, v_0) using the mapping described below.

Once rotation and translation parameters are known, the vehicle's global position can be estimated by chaining the frame-by-frame estimates. A disadvantage of this method is that errors in the parameter estimation accumulate, so that the global position estimate may not be reliable. However, we are interested in the relative short-term motion ($\leq$1s) and the algorithm described above gives good results for this task.

A. Mapping Camera Coordinates to Ground Plane

Figure 2 provides a schematic of the physical setup and geometry in the process of mapping coordinates from the calibrated camera to points on the ground plane. Let us denote the camera coordinates by (u, v) and the corresponding points on the ground plane by (x, y).

From the camera calibration, the focal length, principal point and lens distortion parameters are assumed to be known. We also assume that the height H of the camera above the ground plane and the distance D from the normal of the ground plane, that is going through the camera centre to the point where the principal ray intersects the ground plane, are known.

The tilt of the camera with respect to the ground plane α can then be recovered from

$$tan(\alpha) = \frac{H}{D} \quad . \tag{7}$$

Let v denote the number of pixels in the image's vertical direction to β, measured from the top of the image. The angle β can then be recovered from

$$tan(\beta) = \frac{1}{V}(2v - V)\; tan\left(\frac{VFOV}{2}\right) \tag{8}$$

where V is the vertical dimension of the image in pixels and $VFOV$ is the vertical field of view of the camera system, which is also assumed to be known. From these, we can compute the depth z from the camera to the observed point (x, y) and the distance y on the ground plane from the camera to the observed point (x, y), respectively,

$$y = \frac{H}{tan(\alpha + \beta)} \qquad z = \frac{Hcos(\beta)}{sin(\alpha + \beta)} \quad . \tag{9}$$

From these equations, the apparent motion of a particular image point, e.g. a tracked feature point, can be recovered. The apparent motion generated by such a point on the ground plane causes its observed magnitude in the image plane to change due to its depth z relative to the image plane and its orientation β to the camera axis. From the above equations, we can invert this transform and recover the actual displacement on the ground plane.

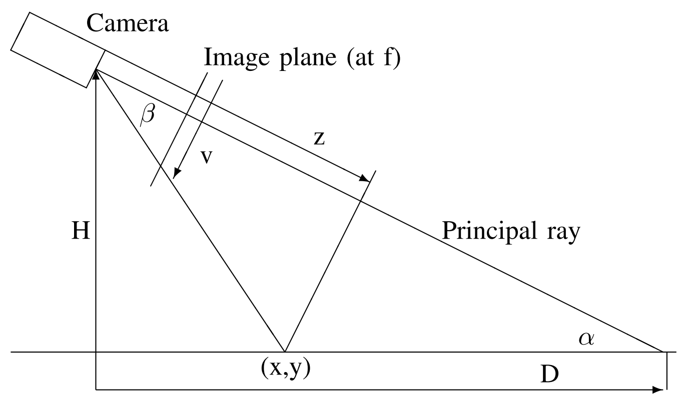

Fig. 2. Mapping camera coordinates to the ground plane.

B. Smoothness Constraint

When using optical flow, such as in [8] and also in our experiments when comparing the FMT approach with previous approaches, visually estimating the displacement on the ground plane from one image to another can be very difficult due to a lack of visually distinctive features on the road surface. Similarly, some points may correspond to a self-moving object, such as other vehicles or pedestrians. If such points are incorporated in the motion parameter estimation, incorrect motion parameters result. If the frame rate is sufficiently high, the estimated parameters should give rise to a smooth motion and the motion observed in the current image should be similar to that in the previous image.

Based on the previous velocity v_{t-1} and the maximum possible acceleration / deceleration a_{max} of the vehicle, it is possible to derive the maximum change in velocity $\Delta v_{max}(t)$ at the current time t

$$\Delta v_{max}(t) = abs(v_{t-1} - a_{max} * \Delta t) \quad . \tag{10}$$

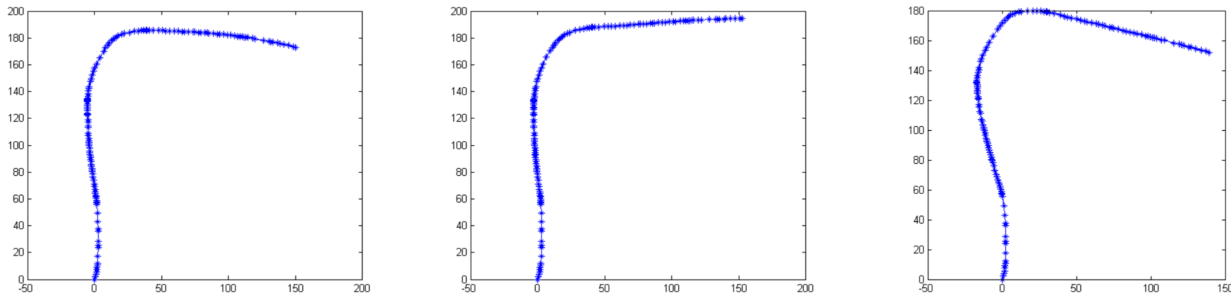

Fig. 3. Resulting paths of vehicle motion on the ground plane for the optical flow method (left), the SIFT + optical flow method (centre), and the proposed FMT method. Units correspond to distances in meters.

Outliers can then be detected if they exhibit a change in velocity from one image to another that is larger than $\Delta v_{max}(t)$. Various methods of treating such outliers are possible. One method is to ignore the motion parameters estimated at time t and to retain the parameters from time $t-1$. This method will be referred to as Method 1 in Section V. Another method limits the change in velocity to $\Delta v_{max}(t)$. This method will be referred to as Method 2 in Section V. Other methods such as smoothing the parameter curve by fitting a bicubic spline function to it are possible. It should be noted that such smoothness constraints are not needed in the FMT approach because it does not rely on individual point correspondences.

V. EXPERIMENTS

For experimental evaluation, we used images from a sequence of 1400 video frames taken by a single camera system in a moving car in an urban environment during daylight (see Figure 1 for an example). The video frames have a spatial resolution of 640×480 pixels and the video frame rate is 60Hz. The sequence contains images of a motion that has a slight left-hand turn at the beginning, followed by a $90°$ right-hand turn about half way through the sequence, before the vehicle is travelling on a straight stretch of road.

Images are first rectified to remove radial lens distortion effects. A window function is applied to remove aliasing effects introduced by the log-polar representation of the FMT. In our work, we tested various window functions and found that a Kaiser window worked best. We used an implementation of the FMT in Matlab on a Pentium IV PC with a 3.4GHz CPU and 2GB RAM. The FMT could be computed in less than 1s per image pair. In rare cases, it was necessary to search the phase correlations for more than one peak which resulted in an increased computational cost. Nevertheless, it is reasonable to assume that a real-time implementation could be achieved, even more so as taking pairs of consecutive video frames is an overly cautious approach. Once the images had been registered, the motion parameters were estimated. For the methods with smoothness constraint, these were applied afterwards.

We compare our FMT approach with a traditional (correlation-based) optical flow method and a method where we first compute SIFT keypoints which are then used as initialisation for an optical flow method. The resulting path of the vehicle on the ground plane is plotted in Figure 3. Note that the paths are very similar initially but diverge after the $90°$ right-hand turn. Differences exist in the amount of rotation recovered by the three methods. The optical flow and FMT methods seem to overestimate the right-hand turn, while the SIFT + optical flow method underestimates it. In the case of the FMT method, this is due to the method currently not computing the image registration parameters with subpixel accuracy. We currently work on an extension of the FMT method that corrects this problem.

Figure 4 shows the resulting egomotion parameters. The translation is not shown because it is directly related to the velocity. The rotation estimates are following the same trend for all three methods, with the FMT showing slightly larger rotation values than the other two methods. For better readability, results for only the first 600 images are shown for the velocity estimates. Of all the methods without smoothness constraint, the FMT had the most coherent velocity estimates. The smoothness constraint Method 1 successfully ignored outliers. In Method 2, a value of $a_{max} = 10m/s$ was used. This method also limited the amount of incoherent velocities but was not as successful as Method 1.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, a novel method for estimating the egomotion of a vehicle from a single camera video sequence has been proposed. It uses the FMT to solve the image registration problem, providing an efficient and accurate way to compute the rotation, translation and scale parameters that describe the relationship between two images in the video sequence. From these parameters and the known camera calibration parameters, the rotation and translation components of the camera motion are derived and, thus, the distance, velocity and path travelled by the vehicle can be estimated.

Unlike other methods, the FMT provides a global method for the image registration problem. This is advantageous both in terms of accuracy as well as computational cost, which is only dependant on the image size and thus can be computed prior to the deployment of the method, allowing for a trade-off between computational cost and accuracy. The proposed method and two variants of optical flow methods have been experimentally tested in this work.

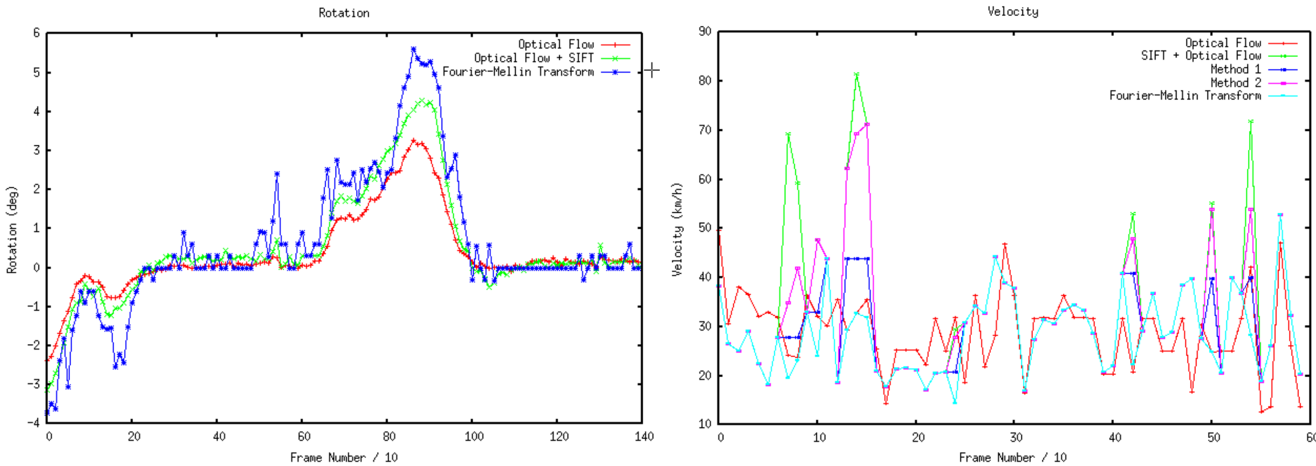

Fig. 4. Resulting egomotion parameters: Rotation angle (in deg) and velocity (in km/h).

In future work, we will compare the results with measurements from a GPS system. We are currently in the process of acquiring such data. Furthermore, we will compare the performance of the proposed method with a recently proposed image registration method that uses log-polar mappings and the Levenberg-Marquardt nonlinear least-squares optimisation algorithm [21]. This latter method has the advantage that it can handle large-scale changes and arbitrary rotation angles for perspective transformations. This in turn would allow us to take larger time intervals between the two images to be registered and thus to lower the computational cost further.

VII. ACKNOWLEDGMENTS

National ICT Australia is funded by the Australian Government's *Backing Australia's Ability* initiative, in part through the Australian Research Council.

REFERENCES

[1] J. Aggarwal and N. Nandhakumar, "On the Computation of Motion from Sequences of Images - A Review," *Proceedings of the IEEE*, vol. 76, no. 8, pp. 917–935, Aug. 1988.

[2] H. Badino, "A Robust Approach for Ego-Motion Estimation Using a Mobile Stereo Platform," in *Proceedings of the 1st International Workshop on Complex Motion IWCM04*, ser. Lecture Notes in Computer Science, B. Jähne, R. Mester, E. Barth, and H. Scharr, Eds., vol. 3417. Günzburg, Germany: Springer, Feb. 2007, revised papers.

[3] E. Dickmanns and B. Mysliwetz, "Recursive 3-D Road and Relative Ego-State Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 199–213, Feb. 1992.

[4] G. Stein, O. Mano, and A. Shashua, "A Robust Method for Computing Vehicle Ego-motion," in *Proceedings of the 2000 IEEE Intelligent Vehicles Symposium IV2000*. Dearborn (MI), USA: IEEE, Oct. 2000, pp. 362–368.

[5] ——, "Vision-based ACC with a Single Camera: Bounds on Range and Range Rate Accuracy," in *Proceedings of the 2003 IEEE Intelligent Vehicles Symposium IV2003*. Columbus (OH), USA: IEEE, June 2003, pp. 120–125.

[6] T. Suzuki and T. Kanade, "Measurement of Vehicle Motion and Orientation using Optical Flow," in *Proceedings of the IEEE Conference on Intelligent Transportation Systems*. Tokyo, Japan: IEEE, Oct. 1999, pp. 25–30.

[7] F. Woelk and R. Koch, "Robust Monocular Detection of Independent Motion by a Moving Observer," in *Proceedings of the 26th DAGM Symposium*, ser. Lecture Notes in Computer Science, vol. 3175. Tübingen, Germany: Springer, Aug. 2004, pp. 27–35.

[8] J. Campbell, R. Sukthankar, I. Nourbakhsh, and A. Pahwa, "A Robust Visual Odometry and Precipice Detection System Using Consumer-grade Monocular Vision," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation ICRA 2005*. Barcelona, Spain: IEEE, Apr. 2005, pp. 3421–3427.

[9] L. G. Brown, "A Survey of Image Registration Techniques," *ACM Computing Surveys*, vol. 24, no. 4, pp. 325–376, Dec. 1992.

[10] B. Horn and B. Schunck, "Determining Optical Flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[11] B. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," in *Proceedings of the 7th International Conference on Artificial Intelligence IJCAI 1981*, Vancouver, Canada, Aug. 1981, pp. 674–679.

[12] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[13] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of the 4th Alvey Vision Conference*, Manchester, UK, 1988, pp. 147–151.

[14] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," in *Proceedings of the Eighth European Conference on Computer Vision ECCV2002*, vol. I. Copenhagen, Denmark: Springer, May 2002, pp. 128–142.

[15] E. de Castro and C. Morandi, "Registration of Translated and Rotated Images Using Finite Fourier Transforms," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, pp. 700–703, Sept. 1987.

[16] B. S. Reddy and B. Chatterji, "An FFT-Based Technique for Translation, Rotation, and Scale-Invariant Image Registration," *IEEE Transactions on Image Processing*, vol. 5, no. 8, pp. 1266–1271, Aug. 1996.

[17] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge, UK: Cambridge University Press, June 2003.

[18] T. Cootes, C. Taylor, D. Cooper, and J. Graham, "Active shape models - their training and applications," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, Jan. 1995.

[19] T. Cootes, G. Edwards, and C. Taylor, "Active Appearance Models," in *Proceedings of the European Conference on Computer Vision ECCV'98*, ser. Lecture Notes in Computer Science 1406, H. Burkhardt and B. Neumann, Eds., vol. 2. Freiburg, Germany: Springer, June 1998, pp. 484–498.

[20] C. Kuglin and D. Hines, "The phase correlation image alignment method," in *Proceedings of the IEEE 1975 International Conference on Cybernetics and Society*. New York, USA: IEEE, Sept. 1975, pp. 163–165.

[21] S. Zokai and G. Wolberg, "Image Registration Using Log-Polar Mappings for Recovery of Large-Scale Similarity and Projective Transformations," *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1422–1434, Oct. 2005.