

Performance of Optical Flow Techniques for Indoor Navigation with a Mobile Robot

Chris McCarthy

Dept. Computer Science & Software Engineering
The University of Melbourne, Parkville,
VIC, 3010, AUSTRALIA
Email: cdmcc@cs.mu.oz.au

Nick Barnes

Autonomous Systems and Sensing Technologies Programme
National ICT Australia, Locked Bag 8001
Canberra, ACT 2601.
Email: nick.barnes@nicta.com.au

Abstract—We present a comparison of four optical flow methods and three spatio-temporal filters for mobile robot navigation in corridor-like environments. Previous comparisons of optical flow methods have evaluated performance only in terms of accuracy and/or efficiency, and typically in isolation. These comparisons are inadequate for addressing applicability to continuous, real-time operation as part of a robot control loop. We emphasise the need for comparisons that consider the context of a system, and that are confirmed by in-system results. To this end, we give results for on and off-board trials of two biologically inspired behaviours: corridor centring and visual odometry. Our results show the best in-system performances are achieved using Lucas and Kanade’s gradient-based method in combination with a recursive temporal filter. Results for traditionally used Gaussian filters indicate that long latencies significantly impede performance for real-time tasks in the control loop.

I. INTRODUCTION

For a number of years there has been interest in the use of optical flow for vision-based mobile robot navigation. Visual motion in flying insects has been shown to be an important cue for navigational tasks such as centred flight in corridors and the estimation of distance travelled [13]. This has motivated new biologically-inspired approaches to mobile robot navigation using optical flow. Behaviours such as corridor centring [11], docking [11] and visual odometry [7] have all been demonstrated using visual motion for closed loop control of a mobile robot. Despite encouraging results, this paradigm has not been broadly adopted in mobile robot research. A perceived lack of robustness in these techniques, and the absence of a systematic approach to their development are likely reasons for this. The choice of optical flow method is an important case in point. The literature does not support any systematic choice of method for mobile robot navigation.

Previous comparisons have assessed optical flow methods on accuracy and/or efficiency, and only in isolation. Since the first major comparison [6], researchers have attempted to address real-world and real-time issues through synthetic image sequences of higher complexity or simple real image sequences with ground truth [1] [10]. Accuracy/efficiency trade-offs have also been examined in [4] over real and synthetic image sequences. Gradient-based optical flow methods generally perform well in accuracy and efficiency comparisons. These comparisons, however, do not adequately support a

choice of technique for mobile robot navigation. Gradient-based methods use spatio-temporal image intensity derivatives, and thus require spatio-temporal filters to pre-smooth the image signal. Temporal filters differ in required frame support, accuracy, and computation times, highlighting the importance of their inclusion in optical flow comparisons. Previous studies have not directly addressed in-system issues or the choice of temporal filters for gradient-based methods. We emphasise the need for comparisons of vision techniques that consider the context of the system.

In this paper we present a comparison of optical flow methods and temporal filters for mobile robot navigation. We focus only on behaviours involving continuous motion such as corridor centring and visual odometry. Continuous motion is characterised by optical flow that is significantly larger than the frame-to-frame changes in the flow on average. This excludes behaviours such as station keeping, where frame-by-frame changes may be larger in comparison with flow magnitude. We aim to provide insight and recommendations for the choice of flow techniques for robot navigation with continuous motion. Three gradient-based methods are included: Lucas and Kanade [8], Horn and Schunck [5], and Nagel [9], as well as Camus’ correlation-based method [2]. Three temporal filters are also included: Gaussian filtering, Simoncelli’s pre-smoothing and derivative filters [12], and Fleet and Langley’s recursive temporal filter [3]. We give an overview and theoretical comparison of these techniques before setting out our methodology for comparison. We then present results from on and off-board comparisons before giving our conclusions.

II. OVERVIEW AND THEORETICAL COMPARISON

Here we introduce the optical flow techniques for comparison and provide theoretical comparisons for robot navigation. Refer to the cited references for full details.

A. Optical Flow Methods

1) *Camus (correlation-based) [2]*: Applies region matching for spatial windows over n previous frames. The window size sets the maximum detectable flow magnitude. The frame support determines the granularity of sub-pixel velocities. We include two versions, both using a 5x5 pixel spatial window: Camus 2 ($n = 2$) and Camus 5 ($n = 5$).

2) *Horn and Schunck (Gradient-based) [5]*: This method combines the well known gradient constraint equation with a global smoothness constraint by iteratively minimising:

$$\int_D (I_x u + I_y v + I_t)^2 + \lambda^2 (|\Delta u|^2 + |\Delta v|^2) dx, \quad (1)$$

where u and v are the horizontal and vertical components of image motion. We initialise velocities using the last computed flow field, allowing sufficient convergence in ten iterations.

3) *Lucas and Kanade (Gradient-based) [8]*: A model of constant velocity is applied on small local neighbourhoods (ω) of the image by minimising:

$$\sum_{\mathbf{x} \in \omega} W^2(\mathbf{x}) ((\nabla I(\mathbf{x}, t) \cdot \mathbf{v}) + I_t(\mathbf{x}, t))^2, \quad (2)$$

where $W(\mathbf{x})$ denotes a window function. Thresholding eigenvalues of the least-squares matrix can be done to improve accuracy however, we did not apply this in our implementation.

4) *Nagel (Gradient-based) [9]*: Second-order image intensity derivatives are used to stop the application of Horn and Schunck’s smoothness constraint over steep intensity gradients. Velocities are obtained through an iterative minimisation. Our implementation feeds forward the previous flow estimate, allowing faster convergence (we use ten iterations).

B. Temporal Filters

1) *Gaussian Filtering*: An isotropic Gaussian filter is applied in convolution for spatio-temporal pre-smoothing. Four-point central differencing (4pcd) is then applied to estimate derivatives. We include two Gaussian filters with standard deviations 1.5 (Gaussian 1.5) and 0.5 (Gaussian 0.5).

2) *Simoncelli’s Matched-Pair Filters [12]*: Simoncelli proposed a filter design for obtaining accurate multi-dimensional derivative estimates using a small low-pass filter and derivative filter. These are related by their simultaneous design and applied as a matched pair through convolution. The implementation used for this paper employs a pre-smoothing step (size three filter) before applying the 5-tap matched-pair filters. We refer to this entire technique as the Simoncelli filter.

3) *Recursive Temporal Filter [3]*: Fleet and Langley proposed a causal temporal filter that is applied recursively, thereby implicitly carrying forward all past frames. Images are filtered via a cascaded implementation of an order n filter, where n is the number of cascades used. A time constant, τ^{-1} , gives the duration of temporal support. In this paper, we use an order three filter ($n=3, \tau^{-1}=1.25$).

C. Theoretical Comparisons

We present a theoretical comparison of the above techniques for robot navigation. We aim to highlight differences that may assist evaluation and discussion in comparisons described later.

1) *Accuracy*: For angular accuracy, Lucas and Kanade is the most accurate method. To achieve real-time performance with Camus, quantisation error is necessarily high, suggesting it is the least accurate method. Horn and Schunck, and Nagel perform relatively poorly in accuracy comparisons over complex scenes. This is due mainly to global smoothing

TABLE I
EFFICIENCY DATA FOR FLOW METHODS.

Method	Time (ms)
Camus 2	460
Camus 5	1770
Lucas & Kanade	263
Horn & Schunck	105
Nagel	630

over regions where this assumption is violated. Nagel handles such regions better via second order derivatives to guide the application of global smoothing.

Reported results for the Simoncelli filter [12] show superior accuracy to the Gaussian 1.5. The recursive filter has been shown to be slightly less accurate on synthetic image sequences [3]. The Gaussian 0.5 will increase noise in derivative estimates which is likely to effect on-board performance.

2) *Efficiency*: Table I and II show efficiency data for all techniques¹. Computation time of filters include time taken to perform filtering and derivative estimation. Storage requirements represent the number of frames that are explicitly stored. Latency indicates the frame delay in response.

Camus achieves low execution time if the search area is small. This yields low maximum detectable flow with high quantisation error. These constraints are significantly restrictive, requiring high frame rates and sufficiently low forward velocity to reduce flow. Nagel is the slowest gradient-based method with second-order derivatives adding significantly more overhead compared with first-order methods. Of the temporal filters, the Gaussian 1.5 requires the largest explicit frame support and frame delay. Given the near equivalent accuracy of the Simoncelli and recursive filters to the Gaussian 1.5, their reduced delay should improve in-system performance.

3) *Robustness*: The constraints imposed on Camus suggests it would be the least robust. High quantisation error increases Camus’ sensitivity to noisy motion fluctuations. The large temporal support of the Gaussian 1.5 and recursive filter (implicitly), suggests these should be less sensitive to such noise. The Simoncelli and Gaussian 0.5 filters may exhibit higher sensitivity due to their reduced frame support.

Nagel may lack robustness due to high noise sensitivity in second order derivative estimation and latency. However, Nagel, and Horn and Schunck are well suited to near constant ego-motion in planar environments (such as corridors) due to their use of global smoothness constraints. Lucas and

¹Times taken on an Intel x86 866 MHz machine.

TABLE II
EFFICIENCY DATA FOR TEMPORAL FILTERS.

Filter	Time (ms)		Support (frames)	Latency (frames)
	1st order	2nd order		
Gauss 0.5	116	237	9	4
Gauss 1.5	170	288	15	7
Recursive	110	177	3	3
Simoncelli	106	209	7	3

Kanade should exhibit sufficient robustness, with relatively fast execution time and accurate flow estimation.

4) *Responsiveness*: Camus has no frame delay and so should exhibit high responsiveness to environmental change. However, Camus is dependent upon flow not exceeding maximum detectable levels. Of the gradient-based methods, Nagel should be the least responsive due to high execution time and second order derivative estimation.

The recursive filter, unlike the Gaussian and Simoncelli filters, implicitly carries forward all past frames. This may inhibit the filter’s immediate response to image motion changes. Less frame support should reduce temporal cohesion to a shorter time period, thereby increasing responsiveness to change. This suggests the Simoncelli and Gaussian 0.5 filters should be the most responsive filters. However, increased noise levels may also be experienced, resulting in a reduction or loss of signal due to noise levels overwhelming the flow field.

III. METHODOLOGY

In this section we present two navigational behaviours implemented for this comparison: corridor centring, and visual odometry. We outline the methodology and performance indicators used in our comparison of optical flow techniques.

A. Corridor Centring

Corridor centring, inspired by observations in honeybees [13], can be achieved by differencing average flow magnitudes in the outer thirds of the image from a forward looking camera:

$$\theta = \tau_l - \tau_r, \quad (3)$$

where τ_l and τ_r are the average flow magnitudes in the left and right peripheral views respectively. θ can be directly used for directional control.

Given constant motion and a straight corridor, the flow field response should exhibit consistent average flow magnitude. The robot should be free of short period directional oscillation resulting from noise introduced through the robot’s ego-motion. Frequent and current flow updates are needed to maintain behaviour stability. Long period directional oscillation through reduced responsiveness is the likely side effect of such latencies. Off-board comparisons over an image sequence depicting near constant motion of the camera can examine temporal cohesion. On-board trials in a static corridor can demonstrate the level of stability in robot directional control.

B. Visual Odometry

Distance travelled can be estimated by accumulating image motion in the peripheral regions of the image over time. At a discrete time t , the visual odometer, d_t , is given by [7]:

$$d_t = \sum^t \frac{4}{\left[\frac{1}{\tau_l} + \frac{1}{\tau_r}\right]}. \quad (4)$$

Odometry estimates will vary in different environments for the same distance. In the same environment, however, the estimate should be repeatable. To compare methods, variance in average distances travelled can be examined for multiple

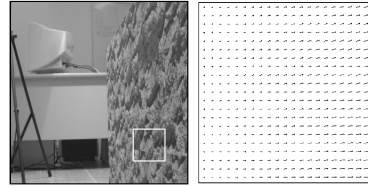


Fig. 1. Frame and flow field (GT) for boxed region from side wall sequence.

on-board trials in the same environment. On-board trials are subject to oscillatory directional control, lateral drift, and environmental changes. To account for such in-system influences, off-board performances can also be examined.

A real image sequence with ground truth allows a quantitative comparison. If the distance measure d_t is repeatable, we expect it to differ by only a scale factor, s , from a ground truth visual odometer g_t . This scale factor should remain approximately constant over time such that:

$$s = \frac{g_t}{d_t} = \frac{g_{t-1}}{d_{t-1}} = \dots = \frac{g_1}{d_1}. \quad (5)$$

IV. OFF-BOARD COMPARISONS AND RESULTS

Off-board comparisons used a real image sequence (Figure 1). This depicts a wall moving in a near parallel direction to the optical axis of the camera. The velocity of the wall’s motion is approximately 5mm per frame. Wall motion is subject to small fluctuations throughout the sequence. Ground truth flow fields were generated for the entire sequence by calibrating the camera and employing a projective warping technique described in [1]. We present these experiments and results below.

A. Corridor Centring

Figure 2 shows average flow magnitudes obtained for each technique across the sequence. No latencies are accounted for. This allows a direct comparison of flow magnitude consistency across the sequence. All methods exhibit variance in flow magnitude in-line with ground truth. All gradient-based methods perform similarly, exhibiting superior consistency to Camus. Camus 2 and 5 both exhibit high sensitivity to small motion changes.

Figure 3 presents average flow magnitudes for all temporal filters with no delays accounted for. Due to space constraints, only results for Lucas and Kanade are shown, however, all gradient-based methods showed similar responses for the same filters. The choice of filter was found to significantly effect performance. The most temporally cohesive responses were achieved using the recursive and Gaussian 1.5 filters. The Simoncelli filter performs slightly worse with marginally sharper fluctuations. This is most evident between frames 19 and 20, and frames 28 to 30. The Gaussian 0.5 exhibited the sharpest fluctuation between frames.

It is important to also consider frame delays in these results. Figure 4 shows the same data with frame delays included. The larger temporal delay of the Gaussian 1.5 response is clearly evident against all other filters.

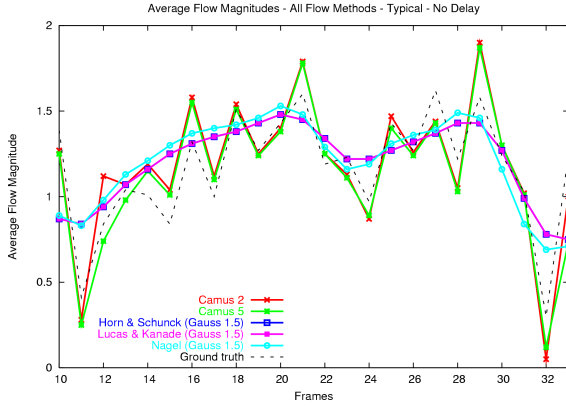


Fig. 2. Average flow magnitudes for all flow methods without latencies.

TABLE III
VISUAL ODOMETRY ERROR ANALYSIS.

Method	Filter	$av(s)$	$\sigma(s)$	$\sigma(s_{av})$
Camus 2	–	1.04	0.06	0.11
Camus 5	–	0.97	0.06	0.13
H & S	Gauss 1.5	1.04	0.10	0.12
Nagel	Gauss 1.5	1.00	0.11	0.12
L & K	Gauss 1.5	1.06	0.10	0.12
L & K	Gauss 0.5	0.87	0.14	0.14
L & K	Recursive	1.07	0.09	0.11
L & K	Simoncelli	1.03	0.11	0.13

B. Visual Odometry

Visual odometry comparisons were conducted using the same side wall sequence. At each frame, the average flow magnitude was added to the accumulating visual odometer.

Table III shows average scale factor errors ($av(s)$) and standard deviations (σ) of scale factor errors for each technique when compared with ground truth. Scale factor errors were calculated using the value of the ground truth visual odometer at each corresponding odometer update.

According to (6), s should ideally remain constant for the distance estimate to be accurate. All methods and filters, with the exception of Camus, exhibit similar levels of deviation

from their average scale factor error ($av(s)$). Camus shows significantly less variance in scale factor error. The last column in Table III shows variances of scale factor error when calculated from the straight line approximation to the ground truth visual odometer (s_{av}). All techniques yield similar results. More variation in results is seen across the temporal filters, though results remain close. The recursive filter shows marginally less variance in s . The Gaussian 0.5 exhibits the most deviation over the sequence on both metrics.

C. Discussion

Gradient-based methods appear more suitable for corridor centring. They exhibit responsiveness to flow magnitude variance, but a resistance to small fluctuations. Camus showed high sensitivity to these fluctuations, suggesting it would not perform as well in the control loop. The recursive filter and Gaussian 1.5 filter both showed near equivalent consistency. Reduced latency with the recursive and Simoncelli filters suggests better in-system performances from these.

Visual odometry results suggest no clear distinctions between techniques. Table III shows both versions of Camus giving significantly lower variances in s . Camus' sensitivity to motion fluctuations gives it a higher correlation with ground truth, explaining the strong performance on this metric. This high sensitivity is likely to impede distance estimates due to motion fluctuations caused by robot ego-motion. Temporal filters yield less sensitivity to motion fluctuations, and so are likely to produce more repeatable distance estimates when the underlying motion is smooth and continuous.

V. ON-BOARD COMPARISONS AND RESULTS

All techniques were integrated into the robot control software, running on an Intel x86 866MHz PC with radio link to a mobile robot. A forward-facing on-board camera was tethered to the PC. Frames from the camera were sub-sampled to 192x144 pixels, with a frame rate of 12.5 frames/sec. Robot tracking was achieved using a calibrated overhead camera.

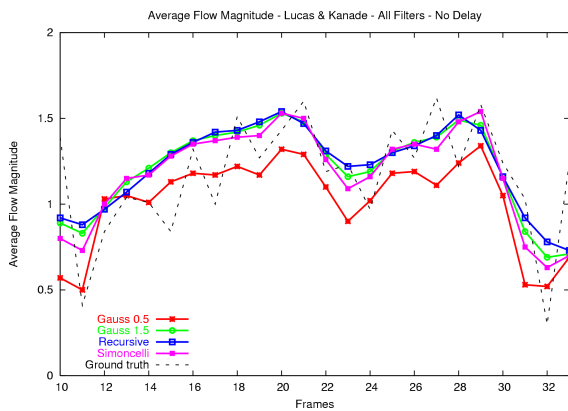


Fig. 3. Average flow magnitudes for all temporal filters without latencies.

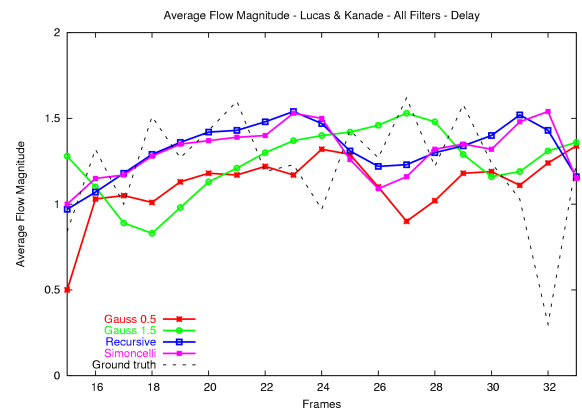


Fig. 4. Average flow magnitude for all temporal filters with temporal delay.

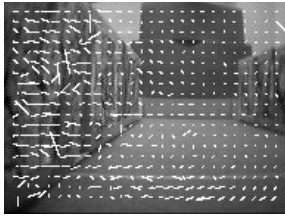


Fig. 5. Sample on-board frame and flowfield (L&K rec).

TABLE IV

ON-BOARD CENTRING AND VISUAL ODOMETRY RESULTS.

Method	Filter	centring error (cm)	flow updates	avg dist (cm)	std dev (avg dist)
Camus 2	–	8	34	253	6
Camus 5	–	11	10	269	12
L & K	Gauss 0.5	4	38	217	7
L & K	Gauss 1.5	6	35	220	10
L & K	Simoncelli	5	38	212	6
L & K	Recursive	3	45	231	3
H & S	Gauss 0.5	7	64	231	12
H & S	Gauss 1.5	9	55	231	13
H & S	Simoncelli	8	65	227	6
H & S	Recursive	5	75	220	8
Nagel	Gauss 0.5	6	19	267	12
Nagel	Gauss 1.5	6	19	227	8
Nagel	Simoncelli	7	20	215	12
Nagel	Recursive	8	19	203	8

A. Corridor Centring

Trials were conducted for each method and filter using a straight corridor, approximately 2.5 meters in length. The width of the corridor was approximately 60 cm. Only directional control was used, with forward velocity kept constant at $0.15m/s$ for all trials. For each filter, a hand-tuned proportional control scheme was used for directional control. Figure 5 shows a sample on-board frame and flow field. Table IV shows average deviations from corridor center (centring error) for each method-filter combination. Figures 6 and 7 show typical path plots for each method and each filter (with Lucas and Kanade) respectively. Table IV clearly shows Lucas and Kanade with the recursive filter to be the strongest combination. This is also confirmed in overhead path plots. Horn and Schunck behaved similarly but generally slightly worse with the same filters. Nagel and Camus struggled to maintain stability of control, both registering failed attempts (not included in Table IV results). Camus 5 performed the worst, showing a lack responsiveness to the approaching wall.

Of the filters, best overall results were achieved by the recursive filter which maintained a centred path and no observed oscillation. The Simoncelli filter also performed comparatively well with first-order gradient-based methods. Centring errors for these filters show a significant drop in performance with Nagel, while the Gaussian filters maintain reasonably similar results for all methods. Path plots for the Gaussian 1.5 showed the largest amplitude of oscillation in the corridor. The Gaussian 0.5 showed less oscillation, however it registered numerous fails in the corridor.

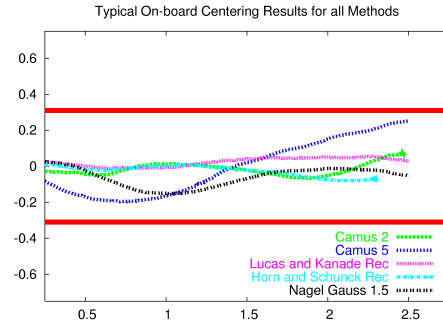


Fig. 6. Typical on-board centring performances for all methods.

B. Visual Odometry

On-board visual odometry trials were conducted using a straight corridor and centring behaviour for directional control. Five trials were conducted for each filter and method. In each trial the robot moved down the corridor until the accumulated visual motion exceeded a preset threshold.

Table IV shows odometer (flow) updates, average distance travelled, and variances for each filter. The best results were achieved by Lucas and Kanade with the recursive filter. Notably, update frequency varies significantly between methods, with better performances coinciding with higher update counts.

C. Discussion

Gradient-based methods generally performed well. Nagel, however, registered fails with all filters, a likely result of slow execution time causing delay in response. The Gaussian 1.5 filter produced long phase, high amplitude oscillatory control in the corridor. Comparing this with the Simoncelli and recursive filters suggests that the larger temporal delay of the Gaussian 1.5 filter is a likely cause for its instability. The poor performance of Camus in corridor centring also confirmed off-board results. Camus 5 was the slowest executing of all methods and so lacked update frequency. Camus 2, while significantly faster, also appeared to lack stability in control.

To further examine Camus' poor performance, frames from the on-board camera were captured and buffered while the robot was manually driven through the corridor. The plotted

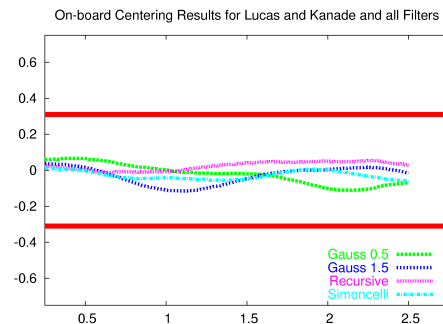


Fig. 7. Typical on-board centring performances for all filters with L&K.

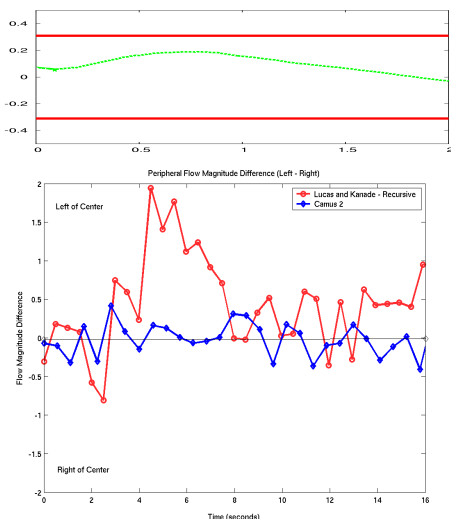


Fig. 8. Peripheral flow magnitude difference for plotted course of robot.

path of the robot is given in Figure 8. Camus was then run in the system with the buffered frames replacing the camera feed. At each iteration of the control loop, peripheral average flow magnitude differences (i.e. $\tau_l - \tau_r$) were logged to examine responses to the approaching wall. These are also shown in Figure 8, along with the responses logged when using the same sequence with Lucas and Kanade. A significant lack of response is shown by Camus when the robot is close to collision. Peak responses shown by Lucas and Kanade indicate that flow is greater than 2 pixels/frame, exceeding Camus' spatial support. Camus requires increased update frequency to achieve stable control and avoid this problem.

Visual odometry results suggest the best choice is Lucas and Kanade with the recursive filter. Techniques with high frequency updates generally performed better. The Gaussian 0.5 and Camus 2 both performed better than expected, a likely result of higher update frequencies and reduced delay. In some cases the Gaussian 0.5 out performed the Gaussian 1.5. The larger temporal delay of the Gaussian 1.5 appears to have caused larger oscillation in the corridor, thereby effecting distance estimation. Camus 5 and Nagel appear to be impeded by low frequency odometer updates, causing over-estimation of target distance. Oscillatory directional control may also have effected visual odometer accuracy. The strong performance of the recursive filter supports this, exhibiting little oscillation in control and high update frequency. The strong performance of less accurate techniques indicates a trade-off of computation speed and accuracy in real-time performance. A loss of accuracy in flow estimation for increased response time can improve overall system performance. This highlights the importance of in-system comparisons for these techniques.

VI. CONCLUSION

In this paper, we have presented results for the comparison of optical flow methods and temporal filters for mobile robot

navigation. We have emphasised the need for comparisons of vision techniques that consider the context of a system. Results were presented for on and off-board trials of two navigational behaviours: corridor centring and visual odometry.

Over all comparisons conducted, the strongest performances were achieved using Lucas and Kanade and the recursive filter. Strong in-system performances were also achieved using Horn and Schunck, and the Simoncelli filter. Camus showed a lack of robustness and responsiveness to large flow. With reduced image size and higher frame rates, Camus should perform better. Nagel's slow execution time appeared to reduce responsiveness in directional control and impede visual odometry.

The recursive filter appears to be the strongest performing filter when integrated into the control loop. Short latency and large implicit temporal support appear to be the reasons for this. For behaviours like docking where responsiveness is of higher priority, the Simoncelli filter might be the better choice. The Gaussian 1.5 filter oscillated significantly more than others in on-board trials, a likely result of its large temporal delay.

ACKNOWLEDGEMENTS

The authors would like to thank Professor John Barron, at the University of Western Ontario, Canada, for providing the implementation of Simoncelli's, matched-pair 5-tap filters.

REFERENCES

- [1] D. Crannitch B. McCane, K. Novins and B. Galvin. On benchmarking optical flow. *Computer Vision and Image Understanding: CVIU*, 84(1):126–143, October 2001.
- [2] T. Camus. Real-time quantized optical flow. *Real-Time Imaging*, 3(2):71–86, April 1997.
- [3] D J. Fleet and K. Langley. Recursive filters for optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(1):61–67, January 1995.
- [4] M. Herman H. Liu, T-H. Hong and R. Chellappa. Accuracy vs. efficiency trade-offs in optical flow algorithms. *Computer Vision and Image Understanding*, pages 271–286, 1998.
- [5] B K P. Horn and B G. Schunck. Determining optical flow. *Artificial Intelligence*, 13(1-3):185–203, January 1981.
- [6] D.J. Fleet J L. Barron and S.S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, Feb 1994.
- [7] S. Venkatesh K. Weber and M V. Srinivasan. Insect inspired behaviours for the autonomous control of mobile robots. In *Proceedings of the 13th International Conference on Pattern Recognition*, volume 1, pages 156 – 60, Vienna, Austria, August 1996.
- [8] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of DARPA Image Understanding Workshop*, pages 121–130, 1984.
- [9] H H. Nagel. On the estimation of optical flow: relations between different approaches and some new results. *Artificial Intelligence*, 33(3):299–324, November 1987.
- [10] M. Otte and H-H. Nagel. Estimation of optical flow based on higher-order spatiotemporal derivatives in interlaced and non-interlaced image sequences. *Artificial Intelligence*, 78(1-2):5–43, November 1995.
- [11] J. Santos-Victor and G. Sandini. Divergent stereo in autonomous navigation: From bees to robots. *International Journal of Computer Vision*, 14(2):159 – 77, March 1995.
- [12] E P. Simoncelli. Design of multi-dimensional derivative filters. In *First Int'l Conf on Image Proc*, volume I, pages 790–793, Austin, Texas, November 1994. IEEE Sig Proc Society.
- [13] M V. Srinivasan and S. Zhang. Visual navigation in flying insects. *International Review of Neurobiology.*, 44:67–92, 2000.