Techniques for Evaluating Optical Flow for Visual Odometry in Extreme Terrain

Jason Campbell [1,2]
jason.d.campbell@intel.com

Rahul Sukthankar [1,2]
rahul.sukthankar@intel.com

Illah Nourbakhsh [2,3]
illah@cs.cmu.edu

[1]Intel Research Pittsburgh
Pittsburgh, PA USA

[2]Robotics Institute, Carnegie Mellon
Pittsburgh, PA USA

[3]NASA Ames Research Center
Moffett Field, CA USA

***Abstract*—Motion vision (visual odometry, the estimation of camera egomotion) is a well researched field, yet has seen relatively limited use despite strong evidence from biological systems that vision can be extremely valuable for navigation. The limited use of such vision techniques has been attributed to a lack of good algorithms and insufficient computer power, but both of those problems were resolved as long as a decade ago. A gap presently yawns between theory and practice, perhaps due to perceptions of robot vision as less reliable and more complex than other types of sensing. We present an experimental methodology for assessing the real-world precision and reliability of visual odometry techniques in both normal and extreme terrain. This paper evaluates the performance of a mobile robot equipped with a simple vision system in common outdoor and indoor environments, including grass, pavement, ice, and carpet. Our results show that motion vision algorithms can be robust and effective, and suggest a number of directions for further development.**

Keywords: robot vision; motion vision; visual odometry; optical tracking; robot motion capture; optical flow.

I. INTRODUCTION

The benefits of reliable navigation by an autonomous mobile robot include the ability to perform a wider range of tasks, operate safely near humans, and move unsupervised through unstructured areas. In contrast, the consequences of poor navigation can range from perceptual confusion and mission failure to collisions, robot damage, or human injury. Vision has the potential to offer low-latency, high-frequency, wide field of view estimates of a robot's environmental state (such as its velocity, relative position, and obstacle/precipice locations) in a compact, passive sensor. Crucial to the increased use of vision sensing are good measurements and procedures for evaluating visual navigation performance.

Our objective in this work is to propose criteria and an experimental method for rigorously evaluating visual odometry systems in real, unstructured environments. This work is a prelude to our own use of visual odometry in other research (specifically, hybrid fixed and mobile sensor networks), and we believe this work, and the data that we have collected, can be useful to other researchers as well. We demonstrate our approach by evaluating an example vision system at outdoor and indoor sites, and discuss the impact of environmental and design parameters on vision system performance.

We have developed a portable apparatus for obtaining reliable ground truth about a moving robot's pose (location and orientation), and software for simultaneously recording video from the offboard apparatus and an onboard digital camera. Using this equipment at a variety of outdoor and indoor sites we have assembled a library of test video sequences and accompanying ground truth data about robot movements on terrain including grass, pavement, ice, and carpet. We plan to add sand, gravel and rocky terrain soon. These ground-truthed sequences allow fair comparisons to be made between different vision systems in real world conditions. They can also help gauge the performance impact of adjusting vision system parameters (e.g., frame rate, number of points tracked, interest point detector thresholds, camera resolution) and thereby reveal the minimal amount of computational and resolving power required to achieve a given performance level – crucial design information for new robots.

This work is especially timely because, as digital cameras and high performance embedded systems proliferate, the cost of equipping a robot with a visual odometry system is falling rapidly. Already, reasonable quality digital cameras (USB, IEEE 1394, and board cameras) are priced comparably to 1D sonar rangefinders ($20-50). Given the power of commodity markets to drive costs down, the near future may see digital cameras becoming the least expensive of all potential mobile robot sensors. Establishing criteria and benchmark test data as a baseline for comparison can help encourage the development of new visual navigation algorithms and improve the use of established ones. Our video sequences and ground truth data are available online at http://info.pittsburgh.intel-research.net/People/jasonc/vo.
Also, while our tests here have been performed with a small robot (27x33x43 cm, 5 kg) the methods used will readily scale to larger robots and autonomous vehicles.

The remainder of this paper is organized as follows: First, we briefly review related work. Second, we introduce the proposed criteria for evaluating visual odometry systems and explain the rationale behind those criteria. Third, we detail our experimental method for measuring visual odometry system performance. Fourth, we describe the robot and vision system used in our tests. Fifth, we present results based on real-world testing using our proposed method. Finally, we review and conclude.

II. Related Work

A wide variety of techniques for visual navigation using robot-mounted cameras have been described over the past several decades. The use of vision to guide an autonomous mobile robot dates back at least to 1976, when Moravec and Gennery used feature tracking to perform visual servoing/course correction on the Stanford Cart [1]. Visual odometry systems have also been successfully used in high-profile projects such as autonomous aircraft [2] and underwater vehicles [3]. Other researchers suggest that, though a general solution to the structure-from-motion problem could remain challenging, the problem of egomotion determination for vision and manipulation is readily solvable [4][5].

Despite this strong history of good research and bright prospects, the adoption of robot vision techniques, including visual odometry, has been slow. Good studies of some likely components of robot vision systems have been published (for instance [6], for optical flow field estimation), but otherwise implementations have appeared largely as point solutions for particular robots and "logical extensions" of published algorithms. Just as Beauchemin and Barron point out in [5] that optical flow has been well addressed in theory but not in practice, the theory of motion vision has been well researched, but many practical aspects of the work remain to be explored. In this paper we take a step towards demonstrating visual odometry as a practical, reliable mobile robot component by defining navigation-performance-based measurement criteria and developing good benchmark data.

III. Evaluation Criteria

Ultimately, the measure of success for a navigational task is safe arrival at an intended destination. For most situations a robot's ability to correctly execute higher level motion plans is a prerequisite for safe maneuvering, so we recognize this as an important evaluation criterion for visual odometry systems. In particular, safe closed-loop maneuvering requires faithful reporting of position and velocity under real-world conditions. This paper focuses on intuitive, graphical measures of such positioning fidelity, primarily using two-dimensional plots of reported position versus ground truth. We augment these detailed but subjective measures with several aggregate metrics of incremental and cumulative odometric performance. This type of evaluation stands in contrast to other common tests typically applied to vision systems, including analyses of synthetic data, measures of image reconstruction quality (frequently used to evaluate optical flow methods), and operation in laboratory or office conditions.

A. Simulated versus Real Data

Vision algorithms may be evaluated with real or with synthetic data. Synthetic data is tempting because it is easier to ascertain precise ground truth and because inconvenient exogenous factors such as variable lighting, motion blurring, optical imperfections, complex motions, and temporal and spatial aliasing may be ignored. However this sharply limits the applicability of such synthetic test results to real-world application domains such as mobile robotics. For instance, in [5] two optical flow extraction methods work well on synthetic data and yet fail when presented with real image sequences. Real data is more difficult to obtain and more challenging to handle, but offers a much better indication of the potential effectiveness of an algorithm in actual use. An objective of this research has been to develop and publish a library of benchmark video sequences recorded in real-world conditions with corresponding ground-truth information. Such a library should have at least two forms of impact: (1) it facilitates fair comparisons between different algorithms; (2) it permits rapid evaluation of the consequences of changes or particular parameter choices within existing algorithms.

B. Extreme Terrain

Rather than limiting our tests to office/lab environments we have chosen to explore a broad variety of terrain, including grass, sand, gravel, sidewalks, and uneven pavement. Each presents its own challenges: sidewalks lack coarse texture; gravel and sand shift under a robot's wheels; grass is of varying height and obscures the

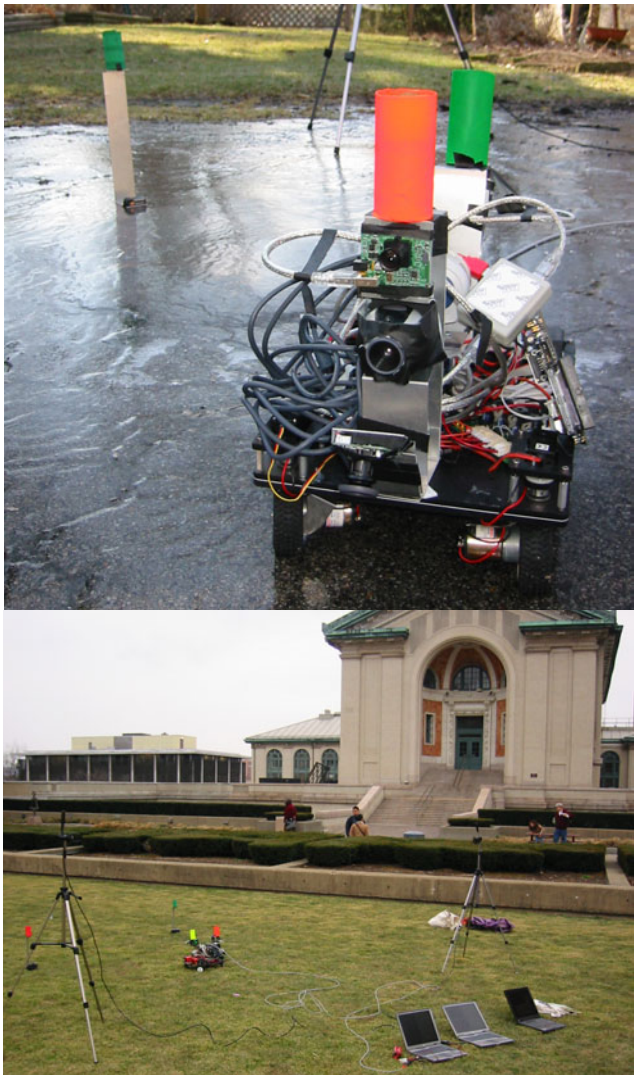

Figure 1. Robots On Ice, In the Grass. Note the tripods for the robot tracking system and the laptop computers used to record video.

true ground plane; ice and asphalt can be highly reflective. A vision system that works well in one terrain may fail in another. Likewise it may be possible to reduce the resource requirements for a given vision system, environment, and level of performance in order to realize financial or computational savings. A broad array of test data compiled in widely different settings is essential for effectively assessing these tradeoffs.

C. *Open- vs. Closed-Loop Test-Patterns*

Some motion-vision algorithms are able to provide quantitative estimates of a camera's egomotion, others can only estimate deviation from a desired goal or path (such as target tracking or corridor following might require). The latter algorithms are less capable than the quantitative ones and must be tested as part of a closed-loop system. That, in turn, precludes exact comparison with other approaches and makes evaluating the consequences of parameter changes much more difficult. This paper focuses only on those methods that can make quantitative motion estimates, and which can be tested using prerecorded, open-loop data.

IV. Experimental Methodology

Obtaining the type of measurements proposed above requires an independent, drift-free means of determining robot pose throughout each experiment. One logical way to achieve such precise ground-truth is to work within a motion capture lab. Using the CMU Graphics Lab motion capture facility we have collected a number of robot motion traces. These data, acquired in a controlled setting with precise ground truth, have enabled us to perform initial analysis of the visual odometry system described and to better understand the problem of evaluating it and similar systems. We have also used data collected in the motion capture facility to validate our own portable robot tracking system. With this portable system we have then been able to collect video and ground truth motion traces in a number of settings which would be difficult or impossible to replicate in a laboratory environment (e.g., sections of lawn, sheets of ice, sand, gravel, asphalt).

A. *Timeline of Each Set of Experiments*

A set of experiments at a given site consists of taking calibration measurements for the robot tracking system, and then recording several (one to a dozen) individual test runs. Each test run involves simultaneous capture of video from an on-robot camera and two tripod-mounted tracking cameras used by the tracking system. Each run takes from 1-5 minutes during which the robot moves 1-5 meters.

B. *Portable Robot Tracking System*

Our portable robot tracking system uses a pair of wide angle digital cameras (f=2.1mm, horizontal field of view exceeding 90°) mounted on tripods arranged so the cameras' fields of view overlap over the entire experimental area. During each robot motion sequence, the cameras record 640x480x24bit color video at 4 or 8 frames per second (fps), depending upon the length and speed of the experiment. The video is transferred via an IEEE 1394 bus to a nearby laptop and is stored uncompressed for off-line analysis. Each video frame is time stamped with the time of acquisition on the laptop (including a lag of approximately 125ms from actual image capture, independent of frame rate).

Analysis of the video streams from the tracking cameras begins by correcting for clock skew between the camera recordings. Because the robot starts each sequence at rest we use the observed start of robot motion as a synchronizing temporal marker to establish a relative time offset for each recording and adjust each frame's timestamp accordingly. This level of timing precision is sufficient to constrain positioning errors resulting from the remaining timing offset to less than 1 cm (4 fps) or less than 0.5 cm (8 fps). After timing correction, we warp each video frame to correct for radial and tangential distortions (which are substantial given the lenses used). We then apply the *camshift* algorithm [7] to track the centroid of each fiducial to sub-pixel precision. Initial color-probability histograms for camshift are established from the first frame of each sequence. This correctly accounts for gradual changes in lighting conditions at the experiment site over time. (such as a rising or setting sun).

During a local calibration phase at each experiment site we measure the pair-wise distances between four specific points on the ground plane and capture images showing four fiducials placed at each of those points (at the same height as the fiducials mounted on the robot). We use those four point locations to define a coordinate frame on the ground and compute an appropriate projective transformation via a homography [15] for each of the two tracking cameras. Those homographies map pixel locations in each camera to locations on the ground plane. By raising the tracking cameras nearly 2m above the ground plane and steeply pitching them down we are able to obtain full two-dimensional location estimates from each camera rather than bearing-only information. Momentary differences in background and lighting may cause the camshift algorithm to lose track of the fiducials, but when an estimate from both tracking cameras is available we combine the two position estimates using a Kalman filter to arrive at a joint, smoothed estimate.

The fiducials themselves have gone through several design iterations, with earlier, smaller ones having proven too small to be seen clearly in tracking cameras' wide angle lenses. The present fiducials are 10 cm long, 4.5 cm diameter cardboard tubes painted with highly saturated colors.

Our tracking system relies on a flat ground plane assumption, and two concerns arise on uneven ground: (1) the positions of the fiducials are incorrectly projected on the ground plane because the calibration homography does not correspond to the robot's local plane; (2) any pitch or roll in the robot's position can potentially cause the estimated position of the centroid in the image to deviate from the true position. For our initial work we have chosen to minimize these sources of error by focusing on terrain which is predominantly flat. Future work may incorporate a different fiducial design and tracking system to permit experimentation in terrain with large elevation changes and deep relief.

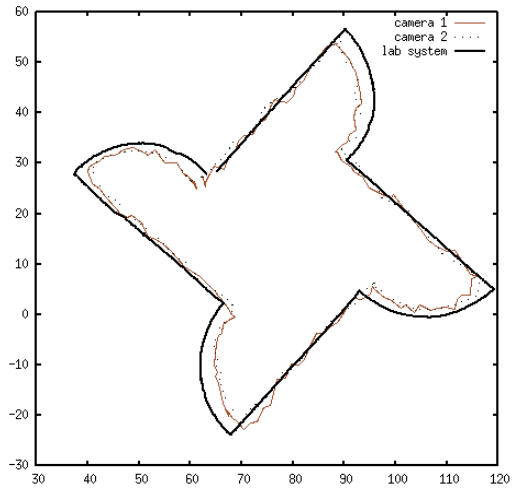

Figure 2. Portable Robot Tracking System Compared To Laboratory Motion Capture System. The lab system's markers were set approx. 3 cm further out from the robot's center of rotation than the portable tracking system's fiducials were, hence the inset of the "camera 1" and "camera 2" tracks during turns. In the motion sequence shown the robot traced a 40 cm square figure, making a turn-in-place at each corner. A quarter-circle arc results at each turn because the fiducials were set at 16.5 and 19.5 cm from the robot's center of rotation.

To validate the portable robot tracking system, we set it up inside the CMU Graphics Lab motion capture facility and recorded robot movements with both the portable tracking system and the laboratory motion-capture system. The lab system was a Vicon Motion Systems model 512 capture unit equipped with twelve high resolution cameras coupled to pulsed, LED-based infrared ring-lights. We used 11 retro-reflective spheres to track the position and orientation of the robot and the positions of the calibration points with millimeter level accuracy. When the 2D results from both systems were compared, we found the portable system to be quite noisy, but on average accurate to within 1-3 cm. Importantly, we observed no cumulative error. (See Fig. 2 for a comparison of the two traces.) The noise in the portable system trace arises from an interaction between the camshift tracker and the de-mosaicing algorithm used in our GRGB-pattern single-chip cameras. The edges of the fiducials appear to change color as they slide across pixel wells on the imaging chip, causing jitter in the estimate of fiducial locations. We are presently considering tracking algorithms that will be more robust to these variations.

C. Robot Platform

As a test platform we have used a small four-wheel-drive, four-wheel-steering robot originally constructed as a prototype for the CMU PER project [8]. This 5kg robot is 27 cm wide by 33 cm long and carries a Cerebellum motor controller board and a single-board computer based on the Intel® XScale® PXA250 processor running at 400 MHz. The robot is capable of arbitrary planar motion and in-place turns as well as Ackerman two- and four-wheel steering. Each 6.3 cm diameter wheel is independently steered and independently driven. There are no wheel encoders, though the drive motors act through

Figure 3. Views From the On-Robot Camera (ice, bright sun on asphalt, grass)

very high reduction gearing (320:1) to provide approximately constant speed robot motion regardless of terrain in the high duty-cycle regime. The stock robot firmware uses this constant-speed operation to perform motor-runtime- and steering-angle-based dead reckoning. Motor speed stability is further enhanced by the use of an onboard DC-DC converter to obtain a steady 17 V for motor power from the gradually discharging 30 V (24 cell NiMH) battery pack. Ground clearance varies between 2.5 and 7.5 cm, enabling operation in rugged terrain.

The on-robot camera used for visual navigation has a focal length of 2.1 mm and is mounted 29 cm above the ground at the robot's front, center edge. In this position, the camera's vertical field of view extends from approximately 45 cm in front of the robot (assuming flat ground) to approximately 40° above the horizon. Other cameras and sensors are also attached to the robot but are not used in this experiment. Because the camera is at the front of the robot, rather than being collocated with the robot's geometric center, the camera experiences mixed translation and rotation during any turn, including "pure rotation"

turns-in-place. Directly above the camera is a large fluorescent orange cylinder (9.4 cm. high, with diameter of 4.5 cm) used as a fiducial for tracking the robot. In a corresponding location and height at the rear center of the robot is a second, differently-colored cylinder. By tracking both fiducials from external cameras, we obtain independent, drift-free estimates of the location and orientation of the robot.

The on-robot camera is connected via an IEEE 1394 bus tether to a nearby laptop PC to facilitate uncompressed video recordings of up to 1GB of data at a time. Typically we operate this camera at VGA resolution (640x480) and capture 8-bits of grayscale information per pixel at 15 fps, though selected movement sequences have also been captured in color and at 30 fps. This video data is then stored using lossless compression to avoid introducing artifacts which could adversely impact vision algorithms.

D. *Test Patterns*

Our test patterns are based predominantly on move-turn-move sequences. Many cause the robot to turn as much as 360° over the course of the test. Incorporating large turns tests the visual odometry system's ability to acquire and track new features as old ones move out of its field of view. For each test sequence, a laptop connected to the robot via an 802.11b wireless network link issues high level commands and on-robot firmware handles motor control and action timing. The robot's wheels are driven at approximately 4 cm/s, and when operated on office carpeting the robot can achieve reasonable short-term translational and rotational accuracy using time-based dead reckoning alone. Accuracy on other surfaces is substantially degraded by wheel slip.

These test patterns may seem to emphasize a separation of translation and rotation (and hence simplify egomotion estimation). However, because the camera on the robot is mounted 16.5 cm forward of the robot's center of rotation, every turn is in fact a simultaneous translation and rotation from the perspective of the vision system. Non-central camera placement, despite being more challenging for the vision system, better represents typical camera deployment scenarios and is especially important for medium-sized and large robots where central camera placement may be difficult, impossible, or result in an unacceptably-restricted field of view.

E. *Why Uncompressed Video? Why IEEE 1394?*

While IEEE 1394 interfaces are not yet common on single-board computers such as might be used to control a robot, this high speed interface offers the most straightforward means of transporting *uncompressed* VGA resolution digital video at reasonable frame rates across a tether. Using uncompressed video in this evaluation is important for two reasons: (a) an embedded video system is most likely to work with uncompressed frames, potentially via a high-speed local interface to the camera (for instance, as in [9]); and (b) some vision algorithms experience degraded performance in the face of compression artifacts. Use of a tether was essential given the large amount of video data captured (over 250 MB/minute for 8-bit grayscale at 15 fps, 3x that for color). Our current experimental apparatus uses a laptop equipped with 2GB main memory to record the video stream from the robot, thus allowing experiments of up to 8 minutes duration at 15 fps, or 4 minutes at 30 fps.

Capturing at the highest frame rate possible has the added advantage of allowing us to decimate the resulting video stream later on to obtain equivalent video at lower frame rates. This is helpful in testing sensitivity of a given algorithm to frame rate. Given that a reduced frame rate translates directly into lower computational demands and lower cost components this is an especially important design variable to investigate.

V. An Example Visual Odometry Algorithm

The visual odometry algorithm we have used for our tests is a simple and direct method based on the assumption that the scene in view is a predominantly flat ground plane at constant, known distance and pitch from the camera, with zero roll. The optical flow field in the portion of the image below a static horizon is used to estimate x and z translation, and the optical flow field above the horizon is used to estimate rotation. To our knowledge this particular method has not been explicitly published, perhaps because of the onerous and restrictive camera geometry assumption required for it to succeed. Nonetheless, this assumption *is* valid for the case of many small mobile robots with no suspension and a rigid frame.

1. The Role of Consensus: Scene motion information tends to incorporate a variety of confusing elements, including tracks of other moving entities, reflections, changing shadows, occlusions, and optical flow field estimation errors. These elements result in outliers which can hide the information needed to estimate camera egomotion. Consensus methods such as RANSAC [10] have long been used when analyzing optical flow fields to resist confusion by outliers and generate estimates across large populations [11]. We use RANSAC methods several times in this algorithm, in both single- and multi-dimensional forms, to condense groups of discrete estimates into consensus estimates used for pose updates.

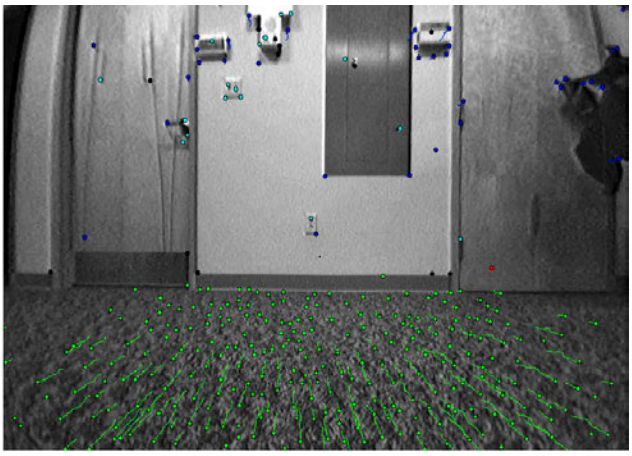

Figure 4. Optical Flow Field Traces, captured in an office environment on carpet. The vectors shown represent 20 frames of motion (slightly over 1 second). Notice the strong motion observed along the ground plane and the relative lack of motion observed above the horizon.

2. Algorithm: Simple trigonometry reveals that, given a plane G obliquely viewed by a perspective camera C with zero roll and yaw and constant pitch relative to G, the projection of the plane into the camera image is particularly straightforward. If the origin of $G(x,z)$ is set as the point directly beneath camera C and the z-axis forms a plane with C's optical axis, and if $C(u,v)$ is set as the intersection of C's optical axis with its the imaging plane, then a feature at (x,z) on G will be imaged at location $(u=k_1 \cdot x/z,\ v=k_2/z)$ in C. We can use this to presumptively map tracked features to locations on the ground plane, and then to calculate corresponding tangential and normal movement distances for optical flow field vectors.

For each frame of video we first un-warp it to remove radial and tangential distortion, then process it with the algorithm described in [12] and [13], using the differences between adjacent frames to estimate the optical flow field.

Given a horizon line measured during a one-time calibration process described below, we divide the flow field vectors into three groups: *sky* for vectors substantially higher than the horizon line, *horizon* for vectors close to the line, and *ground* for vectors substantially below the horizon line.

For the set of flow vectors in the *sky* we retransform the vector endpoints back into original (radially distorted) camera coordinates and use the consensus value of $(u(t)-u(t\text{-}1)) \cdot k_3$ as our estimate $\hat{r}$ of the robot's rotation.

For the set of flow vectors on the *ground*, we apply the perspective transform discussed above to convert each vector in the image frame to a displacement along the ground plane. From those ground plane vectors we then subtract the projected rotation field implied by the estimated rotation, $\hat{r}$. Finally, we choose the consensus displacement vector $[\hat{x}, \hat{z}]^T$ from the remaining ground motion field. This is our estimate of tangential and normal translation. We ignore flow vectors on the *horizon*.

3. Calibration: We first calibrate the camera for radial and tangential distortion [14], and establish the amount of cropping necessary to eliminate vignetting after the distortion correction has been applied. Next we measure the vertical and horizontal field of view of the undistorted, cropped image. Then we mount the camera on the robot and measure the location of the horizon in the camera view. These parameters are assumed not to change and hence need only be measured once at robot assembly.

4. Speed: The code used for the analyses presented here has not been designed to run in real-time. However, we have also implemented a version of this algorithm as part of a demonstration system which performs closed loop, real-time control of a small robot, including visual servoing, course correction, pose estimation, and hazard detection (precipices and obstacles) while simultaneously displaying the video stream being analyzed and overlaying it with an optical flow field and other annotations. The entire system runs at between 8 and 10 fps on a 1.6 GHz Intel® Pentium® III-based laptop computer.

VI. RESULTS

Revealing the specific situations in which a visual odometry system is reliable, versus those where more work is required, is a particular goal of this research. Here we analyze the performance of our example visual odometry algorithm in four different settings. Table I below summaries the results in each setting, while the *x-z* plots on the following page offer a more graphical and detailed depiction of vision system performance. In the *x-z* plots the ground-truth motion of the robot is shown as a thin, dotted, red line, while the motion estimate from the visual odometry system is shown as a thick, solid, black line. All scales are in centimeters and each commanded robot move was a 40 or 80 cm square.

A. Aggregate Measures

The aggregate figures in Table I obscure many of the details of spatial performance better observed in the *x-z* plots, but do reveal a number of interesting points about the overall behavior of our test algorithm:

First, note the dramatic reduction in incremental rotational error in the outdoor tests. We believe this is due to the abundant high-contrast textured regions offered by an outdoor horizon. Because features tracked on such regions lie effectively at infinity, they are ideal for measuring rotation.

Second, note the substantial increase in incremental translational error in the outdoor tests. We believe this is due to terrain-specific issues: Ice exhibits specular reflections which may be tracked as though they were static ground features. Asphalt in the bright midday sun causes glare which leaves few trackable features along the ground plane. Grass presents a complex surface which masks the true ground plane, resulting in consistent overestimation of translational motion. The particular lawn on which we tested also included several small undulations, which lead to substantial errors in estimates of translation in the z dimension (see also Fig. 6).

Third, notice that cumulative error rate, averaged across each entire test, is considerably lower than one might infer from the incremental error measures alone. This is borne

TABLE I. VISUAL ODOMETRY PERFORMANCE BY TERRAIN TYPE

Terrain	Incremental Error, Translation [a]	Incremental Error, Rotation [b]	Average Cumulative Error Rate [c]
Indoors/Carpet	0.3	14.2	0.26
Outdoors/Grass	2.2	4.7	0.41
Outdoors/Asphalt	4.3	5.8	0.49
Outdoors/Ice	3.5	10.5	0.43

a. Expressed in centimeters translation per 24 video frame period, corresponding to 1.6 seconds at 15 fps or approximately 8 cm of robot motion. This number is an average of absolute values and thus penalizes oscillation, unlike the *x-z* plots.

b. Expressed in degrees per 24 frame period, also an average of absolute values.

c. Expressed in centimeters net Euclidian distance per 24 frame period based on the difference between the visual odometry system's final position estimate and the final ground truth position.

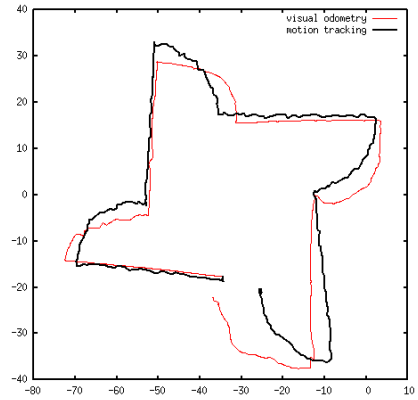

Figure 5. Visual Odometry on Office Carpet

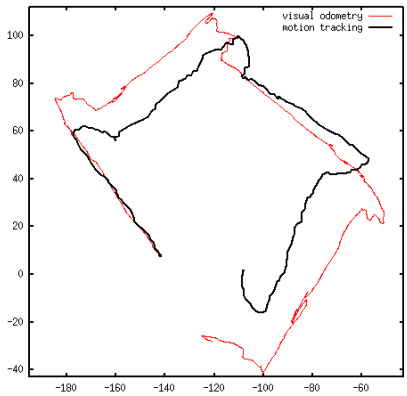

Figure 6. Visual Odometry on Grass

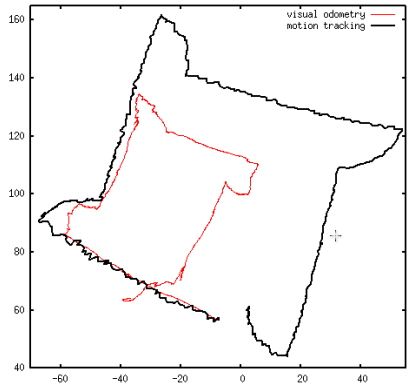

Figure 7. Visual Odometry on Asphalt in Bright Sun

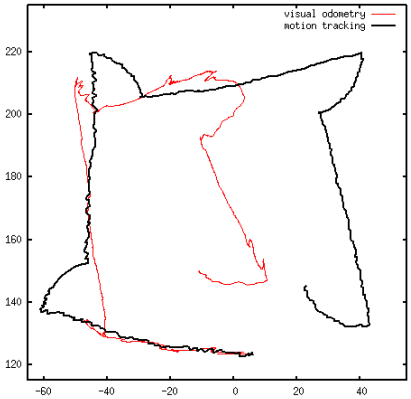

Figure 8. Visual Odometry on Icy Pavement

out in the *x-z* plots in Figs. 5 – 8, which show better overall positioning performance than the rotational error measure in particular would suggest. From studying the detailed motion data and position estimates from each experiment we believe this lower-than-expected cumulative error rate reflects egomotion estimation errors which cancel one another out over time (e.g., changes in camera pitch typically cancel out). Because the incremental error measures shown in Table I are averages of absolute values, errors which cancel out are fully penalized even though they may contribute little to cumulative error. Finally, we also believe the incremental performance measures are higher across the board than they should ideally be due to jitter in the ground truth data, as discussed in Sec. IV.B.

B. X-Z Plots

The *x-z* plots at the left suggest many of the points made in the summary table, but also provide further detail about vision system performance in specific situations. For instance, note that in Fig. 5, executed indoors on carpet, the visual odometry algorithm yields accurate results during lens-axis-parallel translation (e.g., line segments are shown at the correct scale), and also during axis-normal translation (e.g., smooth quarter-circles are shown at each corner, accurately depicting the movements of the offset camera). However, rotational estimates are less accurate. This appears to be because objects in the "sky" (i.e., on nearby walls) are sufficiently nearby to shift in the camera during translational movements as well as rotational ones.

In contrast, note that in a test run outdoors on grass (Fig. 6), lens-axis-parallel translation is overestimated because the surface of the grass is higher than the true ground plane. Thus tracked features on it appear to move further and faster. The section of grass used for this test also exhibited small undulations which appear as apparent doubling-back in the visual odometry trace. When traversing these very shallow hills, the visual odometry system actually estimates that the robot reverses direction.

In Fig. 7, a bright, sunny sky sharply reduces contrast on the pavement in front of the robot. (see also Fig. 3) The result is an optical flow field in the "ground" region of the camera which is too sparse for accurate estimation of translational movements. However, rotational accuracy remains high because the sky regions have many high contrast features. Finally in Fig. 8 specular reflections from the wet surface of the ice complicate estimation of translational velocities while an occasionally blank horizon and sky reduce accuracy when estimating rotational velocity as well.

C. Sensitivity Analyses

Figs. 9 and 10 illustrate the value of recorded data in permitting sensitivity analyses, in this case of the video frame rate supplied to the vision system. Conducting parameter-sensitivity analyses can be a good way to understand the performance envelope of a given vision system. Just as a good software profiling tool can help identify sections of a program where optimization would be especially fruitful, sensitivity-analyses such as these can help to identify aspects of a vision system which might be

improved. In Figs. 9 and 10, for instance, we find that accurately estimating distance traveled while turning can actually be more difficult on carpet than on ice as frame rates are reduced. We hypothesize that this is due to the highly regular appearance of the carpet and the increased potential for temporal-spatial aliasing at low frame rates. This suggests that a change in the optical flow estimation algorithm parameters might better avoid aliasing and improve overall visual odometry performance.

Sensitivity analyses can also be useful during the design and development process for a new robot or vision system. For instance, a designer can determine the minimum camera resolution or frame rate required to achieve a particular level of positioning accuracy, or an engineer might subject a new version of robot firmware to a battery of visual odometry tests before releasing it.

VII. Conclusion

Vision is a uniquely powerful sensing option for mobile robots, and one which will become increasingly attractive in the coming years. As digital cameras and embedded processors targeted at the consumer market become more powerful and less expensive vision may well surpass all other mobile robot sensors in cost effectiveness. Yet despite decades of strong computer vision research many of the more practical aspects of applying vision to mobile robot navigation have received scant attention, including rigorous methods for empirical testing in extreme conditions.

We have described a portable experimental apparatus for obtaining video sequences from a robot-mounted camera and corresponding ground-truth traces of the robot's motion. We have proposed basic evaluation criteria for mobile robot visual odometry systems. Finally, we are releasing a library of video with accompanying ground truth series for use by other researchers in the field.

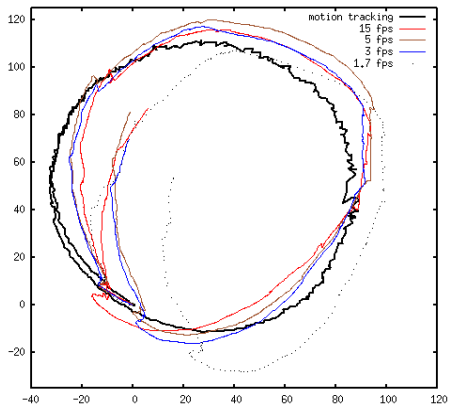

Figure 9. Sensitivity to frame rate – 105 cm diameter circle on ice. The dotted trace which breaks away from the others has been analyzed an effective frame rate of 1.7 frames per second (fps). Above 3 fps, performance of the algorithm plateaus in this particular case.

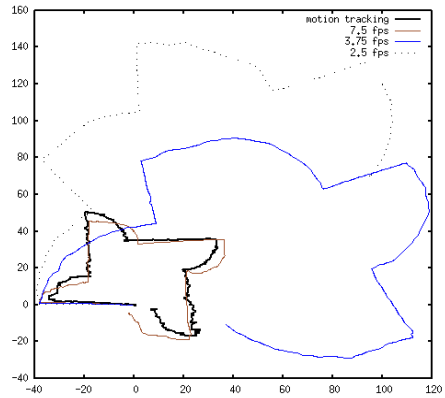

Figure 10. Sensitivity to frame rate – 40 cm square figure on carpet. Only the 7.5 fps trace accurately reproduces the scale and shape of the ground track (thick line). At lower frame rates scale is overestimated.

Acknowledgment

We wish to thank Jessica Hodgins, Moshe Mahler, Mike Stevens, Justin Macey, and the CMU Graphics Lab Motion Capture Facility for their help acquiring ground truth data. We also thank Emily Hamner, Eric Porter, Brian Dunlavey and the CMU Personal Rover Project for their supply of the Personal Exploration Rover prototype.

References

[1] D. Gennery and H. Moravec, *Cart Project Progress Report*, Stanford University, contract NASW 2916, Sept 1976.

[2] O. Amidi, T. Kanade, and J. Miller. "Vision-based autonomous helicopter research at CMU." In *Proceedings of Heli Japan 98*, Gifu, Japan, 1998.

[3] R. Marks, H. Wang, M. Lee, and S. Rock. "Automatic visual station keeping of an underwater robot," in Volume 2, *Proceedings of IEEE Oceans '94*, Pp. 137–142.

[4] C. Tomasi and J. Zhang. "Is Structure-From-Motion Worth Pursuing?" *Seventh International Symposium on Robotics Research*, ISRR '95, October 1995.

[5] S. Beauchemin and J. Barron. "The computation of optical flow," *ACM Computing Surveys*, 27(3):433—467, 1995.

[6] J. Barron, D. Fleet, S. Beauchemin, and T. Burkitt, "Performance of optical flow techniques," *International Journal of Computer Vision*, vol. 12, pp. 43—77, 1994.

[7] G. Bradski, "Computer Vision Face Tracking For Use in a Perceptual User Interface," *Intel Technology Journal*, (2), 1998.

[8] Personal Rover Project web site, unpublished, http://www.cs.cmu.edu/~personalrover/

[9] A. Rowe, C. Rosenberg, I. Nourbakhsh. "A Low Cost Embedded Color Vision System," In *Proceedings of IROS* 2002.

[10] M. Fischler and R. Bolles. "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, 24 (6), 381—395, 1981 .

[11] A. Bab-Hadiashar, "Optic Flow Calculation Using Robust Statistics," in *Proceedings of CVPR* 1997, p. 988-993.

[12] B. Lucas and T. Kanade. "An Iterative Image Registration Technique with an Application to Stereo Vision," *International Joint Conference on Artificial Intelligence*, pp 674-679, 1981.

[13] J.-Y. Bouguet. "Pyramidal Implementation of the Lucas Kanade Feature Tracker," OpenCV Documentation, Intel Corporation, Microprocessor Research Labs, 1999.

[14] R. Tsai, "An efficient and accurate camera calibration technique for 3D machine vision," *Proceedings CVPR, 1986*

[15] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.