

Monocular Visual Odometry for an Unmanned Sea-Surface Vehicle

George Terzakis¹, Riccardo Polvara², Sanjay Sharma², Phil Culverhouse² and Robert Sutton²

Abstract—We tackle the problem of localizing an autonomous sea-surface vehicle in river estuarine areas using monocular camera and angular velocity input from an inertial sensor. Our method is challenged by two prominent drawbacks associated with the environment, which are typically not present in standard visual simultaneous localization and mapping (SLAM) applications on land (or air): a) Scene depth varies significantly (from a few meters to several kilometers) and, b) In conjunction to the latter, there exists no ground plane to provide features with enough disparity based on which to reliably detect motion. To that end, we use the IMU orientation feedback in order to re-cast the problem of visual localization without the mapping component, although the map can be implicitly obtained from the camera pose estimates. We find that our method produces reliable odometry estimates for trajectories several hundred meters long in the water. To compare the visual odometry estimates with GPS based ground truth, we interpolate the trajectory with splines on a common parameter and obtain position error in meters recovering an optimal affine transformation between the two splines.

I. INTRODUCTION

Outdoor visual SLAM is a problem under constant scrutiny in the robotics research community [1], [2], [3], [4]. However, to the best of our knowledge, there have been not many cases in which the SLAM algorithm targets sequences of estuarine and possibly, natural scenes from the vantage point of a surface vehicle. Perhaps the only known recorded case is the work of [5] on lakeshore monitoring which makes use of visual SLAM to localize the vehicle on the surface of the lake. The approach of Griffith et al. relies on the same tools as the method described in this paper. The idea is to detect a sparse set of Harris corners [6] in an image and thereafter compute camera pose by tracking these features using the pyramidal Lucas Kanade tracker [7], [8]. Of course, the primary objective of Griffiths work is the registration of shore images and vehicle localization is simply a subsidiary operation used to confine the search for an image within a subset of images taken in a region close to the estimated position of the boat. It follows that multiple poses can be refined without time limitations in large-scale bundle adjustment runs.

The work in this paper is motivated by the scenario of a surface vehicle [9] cruising autonomously on GPS feedback which may be interrupted; therefore visual odometry on lakeshore images is used as an auxiliary localization system for the vehicle during periods in which GPS reception is

disabled. Thus, in contrast with Griffith et al., this is an exclusively real-time visual SLAM problem and the exact same approach cannot be employed due to the inherent limitations in computational resources. There are however time-efficient alternatives which employ non-linear optimization over only a small pool of poses and images in a sliding time-window, such as local bundle adjustment [10] and generally provide satisfactory pose estimates on demand.

One other significant limitation in our application has to do with extreme depth variations in the shore sequence (1). Although there are ways to assess the nature of the photogrammetric degeneracies in the geometry of two views [11], [12], [13], [14], these methods require Monte-Carlo based model selection and would impose a significant overhead if applied on a regular basis. One remedy would be to include ground-plane features in the field of view [3], [15], [16] and allow motion estimation to be affected primarily by these features. Unfortunately, this is generally not an option when the camera is on-board a surface vehicle due to the presence of the water-surface instead of the ground. It therefore becomes necessary to infer camera motion solely based on the features in the background beyond the sea-surface.

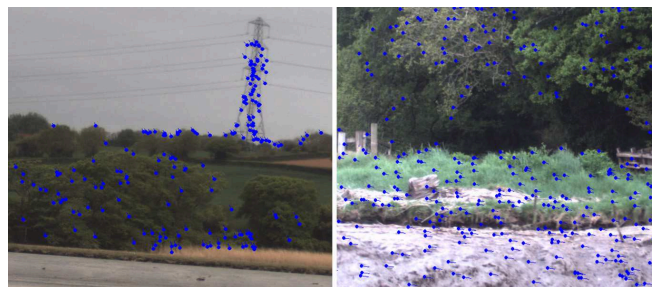


Fig. 1: Extreme opposites in estuarine sequences. On the left, only very distant features over the horizon can be tracked, while on the right, scene depth is very small. In the first case, disparity is very small and only a few carefully selected correspondences can determine motion; in the second case, the disparity is sufficient to determine motion from the majority of features.

Another significant problem that arises from extreme depth variation is the fact that the map can be potentially harmful for pose prediction, due to tracking noise induction with large depth. The problem of reconstructing points with very small disparity is that tracking noise is augmented proportionally to depth as shown in Figure 2. And since having exclusively distant points in the map is a very likely case as illustrated

¹George Terzakis is with the University of Portsmouth, PO1 3HF, UK. Email: george.terzakis@port.ac.uk

²Riccardo Polvara, Sanjay Sharma, Phil Culverhouse and Robert Sutton are with Plymouth University, PL4 8AA, UK. Emails: {riccardo.polvara, sanjay.sharma, p.culverhouse, r.sutton}@plymouth.ac.uk

on the left of Figure 1, it will be difficult to assess the quality of the position estimates in order to choose inliers for pose estimation with a Perspective-n-Pose (PnP) algorithm. In short, the map points obtained from scenes with large scene depth can jeopardize SLAM if used for pose prediction.

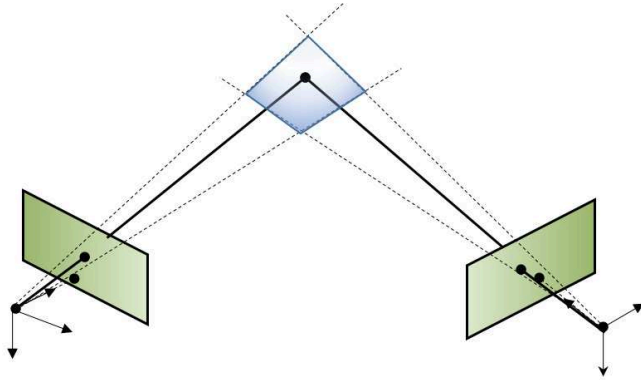


Fig. 2: Uncertainty induction with depth. The size of the blue shaded rhombic region surrounding the triangulated map point is representative of its uncertainty. A similar sketch can be found in Multiple View Geometry in Computer Vision (Hartley and Zisserman 2003).

II. PROBLEM SETUP

A. Overview

The method proposed herein aims at estimating the pose of a sea-surface vehicle in the context of GPS-denied scenario using an on-board camera with view of the shore. In particular, in the case of GPS signal loss, the vehicle should be able to continue cruising autonomously based solely on visual feedback until GPS becomes available again. We focus on river estuarine environments where the primary assumption of shore visibility is met. The camera is mounted on the side of the vehicle as shown in Figure 3.



Fig. 3: The unmanned surface vehicle. The camera (circled in red) is mounted on the left hull, pointing sideways.

B. Apparatus

All video sequences were obtained using a Pointgrey Flea3 USB3 camera, typically at a resolution of 800600 pixels. We addressed camera calibration with Zhangs chessboard

method [17], [18] as implemented in OpenCV computer vision library [19]. For the rest of this paper, it is assumed that the camera is calibrated and all formulas involving projections are given in normalized Euclidean coordinates.

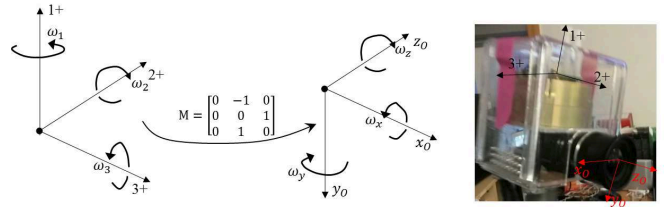


Fig. 4: Flea3 and SiIMU02 (right). The SiIMU02 coordinate frame (left) is shown with the 3 axes labelled 1+, 2+ and 3+ and M denotes the transformation which transforms the sampled angular velocity vectors with the camera frame (shown in red).

Inertial data were sampled using a Goodrich SiIMU02 (Figure 4) at a rate of 150 Hz. It should be noted that the SiIMU02 coordinate frame does not represent a right handed coordinate system and therefore it is necessary to apply a transformation M to the samples in order to obtain the angular velocities x , y , z about the axes x , y , z of the local camera frame. Thus, the rotation matrix representing the change in IMU orientation is:

$$R = I_3 + \frac{\sin \|\omega\|}{\|\omega\|} [\omega]_{\times} + \frac{1 - \cos \|\omega\|}{\|\omega\|^2} [\omega]_{\times}^2 \quad (1)$$

where $\omega = [\omega_x \ \omega_y \ \omega_z]^T$ is the sampled angular velocity vector. We addressed the problem of minor misalignments between the camera the IMU using the recovered camera calibration extrinsic parameters as described in camera-aided robot calibration [20].

III. METHOD

Our method loosely follows the standard SLAM pipeline of pose prediction and refinement with measurements over a sliding window of no more than 5 frames. The idea is to detect new features in each frame and track them for up to 5 frames in the sequence. However, pose is estimated directly on the pairs of corresponding Euclidean projections instead of using the reconstructed map-points. In effect, this is a version of SLAM in which the map can only be inferred from camera pose in conjunction with the measurements and is not directly involved in pose prediction. Figure 5 illustrates our adaptation of the SLAM paradigm in the form of a Bayes network. The pose at time instance- t (time is discrete) is a vector $x_t = [s_t \ \psi_t]^T$ containing the position of the camera s_t and its orientation parameter vector ψ_t . We also denote the IMU angular velocity sample with ω_t and the normalized Euclidean projections of the tracked features with m_t .

In the SLAM adaptation of Figure 5, each tracked feature location at time t is regarded as a function of its projected location and camera pose in the view in which it was

corresponding to the projections in x and y axes:

$$(x'_2 - x_1)Z_1 = x'_2 b_z - b_x \quad (6)$$

$$(y'_2 - y_1)Z_1 = y'_2 b_z - b_y \quad (7)$$

where $m'_2 = [x'_2 \ y'_2 \ 1]^T$, $m_1 = [x_1 \ y_1 \ 1]^T$ and $b = [b_x \ b_y \ b_z]^T$. The depth can be eliminated if we simply multiply eq. 6 by $(y'_2 - y_1)$ and eq. 7 by $(x'_2 - x_1)$ and then, subtract eq. 7 from eq. 6:

$$-(y'_2 - y_1)b_x + (x'_2 - x_1)b_y + ((y'_2 - y_1)x'_2 - (x'_2 - x_1)y'_2)b_z = 0 \quad (8)$$

We therefore obtain a linear equation in the components of b which can be used to formulate an overdetermined linear system that can be solved in ordinary least squares fashion. It should be stressed here that eq. 8 is tolerant to points with very low disparity as it is quite evident that, if both terms $(x'_2 - x_1)$ and $(y'_2 - y_1)$ vanish, then the equation becomes trivial ($0b = 0$) and subsequently has little or no effect on the least squares estimate. We may now formally "repackage" eq. 8 for the i^{th} feature in a new camera view at time t using a determinant:

$$\det \begin{bmatrix} R_{h_i}(s_t - s_{h_i}) & (m_t^{(i)})' - m_{h_i}^{(i)} & (m_t^{(i)})' \end{bmatrix} = 0 \quad (9)$$

where h_i is the time index of the camera view in which the i^{th} feature was originally detected, s_t and s_{h_i} are the position of the camera at times t and h_i in world coordinates, R_{h_i} is the orientation matrix of the camera at time h_i , $m_{h_i}^{(i)}$ is the normalized Euclidean projection of the feature in its home camera view and $(m_t^{(i)})'$ is the unrotated normalized Euclidean projection in the camera view captured at time t .

In the initializing pair of views (i.e., for $h_i = 0$ and $t = 1$), the overdetermined linear system will obviously be homogeneous and therefore the solution will be the direction of the baseline between the respective camera frames. In this case, of multiple camera views, the solution will be sign-ambiguous and therefore it will be necessary to reconstruct the scene and vote for the best reconstruction (i.e., the one with fewer negative/vanishing depths). However, in the case of more than two camera views, the system becomes a standard least squares formulation with a unique scale-aware solution in which scale is infused by the known positions of previous camera frames.

B. Outlier rejection

To eliminate outliers we employ a robust random sample consensus [22] scheme called MLESAC proposed by Torr and Zisserman [23]. For each new camera view, we compute the baseline using eq. 9 over randomly picked minimal subsets of at least 3 points. Then, for the recovered camera position we obtain a robust epipolar score on the misalignment of epipolar planes between camera views. Unlike the case of Sampson error [24], our epipolar misalignment error works directly on the relative pose without the need to use an essential matrix and it is generally easy to compute.

Consider the two corresponding normalized Euclidean projections m_1 and m_2 (again, superscripts are dropped for

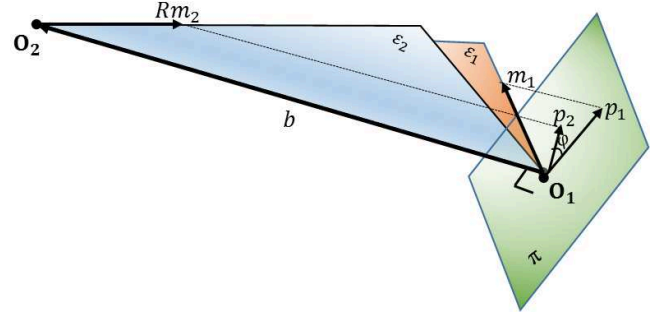


Fig. 7: Measuring misalignment of epipolar planes ϵ_1 and ϵ_2 on a plane π , orthogonal to the baseline b . The misalignment angle cosine will be the inner-product of the normalized projections of m_1 and Rm_2 on the plane π .

simplicity) in two camera views indexed by 1 and 2. Then, if the correspondences are noisy, the two epipolar planes defined by m_1 , m_2 and the baseline will not coincide as shown in Figure 7 and will subsequently form an angle between them. Since the baseline vector b is common for both planes, one way of obtaining the angle between the two misaligned planes is to project the two correspondences onto the orthogonal space of the baseline, which is essentially a plane π through the origin O_1 and to which normal is the baseline. The (projector) matrix P that projects a vector onto the space orthogonal to b is:

$$P = I_3 - bb^T \quad (10)$$

Thus, the projection of m_1 would be:

$$p_1 = Pm_1 = (I_3 - bb^T)m_1 \quad (11)$$

The projection of m_2 , accounting for the difference in the orientation of camera frames, is:

$$p_2 = PRm_2 = (I_3 - bb^T)Rm_2 \quad (12)$$

where R is the relative orientation matrix. The cosine of the angle between the two planes $\epsilon_1 = \langle m_1, b \rangle$ and $\epsilon_2 = \langle Rm_2, b \rangle$ will be the following inner-product:

$$\begin{aligned} \cos \phi &= \frac{p_1^T p_2}{\|p_1\| \|p_2\|} \\ &= \frac{m_2^T R^T (I_3 - bb^T) m_1}{\sqrt{\|m_1\|^2 - (m_1^T b)^2} \sqrt{\|m_2\|^2 - (m_2^T R^T b)^2}} \end{aligned} \quad (13)$$

Clearly, the measure of eq. 13 is a score that increases with the accuracy of the correspondence and therefore we seek to maximize it. Nominal values used for the MLESAC cutoff bound were in the range of $\cos 7^\circ$ to $\cos 3^\circ$. It should be stressed here that the score of eq. 13 penalizes angles above 90° (i.e., correspondences with negative depth in exactly one view), whereas the classic epipolar constraint does not (due to the minimization requirement). Most importantly, it is a score that is independent of the type of camera motion and fitted model (homography or essential matrix), including the

case of pure rotational camera motion, in which case the two projection directions should be collinear and the formula reduces to a simple inner-product which yields the cosine of the angle between m_1 and Rm_2 .

C. Iterative refinement

Pose estimates are iteratively refined with the Gauss-Newton method over a few (usually 3) recent views. Instead of the traditional reprojection error [24], we employ an epipolar alignment score, similar to the one of eq. 13. In particular, instead of considering the misalignment of epipolar planes, this time, we focus the angle between the direction of the second correspondence and the normal of the first plane. This way, a minimization problem is obtained, suitable to apply the Gauss-Newton method. Suppose a sliding window of n camera views is used. We construct the following cost function C over this window:

$$C = \sum_{k=t}^{t+n-1} \sum_{i \in V_t} \left(m_i R_t^T [s_t - s_{h_i}]_{\times} R_{h_i} m_{h_i} \right)^2 \quad (14)$$

where V_t is the set of visible points in the window commencing at time t , h_i is the time index of the "home" view (i.e., first detection) of the i^{th} feature, $m_i^{(i)}$ is the normalized Euclidean projection of the i^{th} feature in the camera view at time t and R_k, s_k are the orientation matrix and position vector (in world coordinates) of the camera at time $t = k$.

The cost function of eq. 14 is a sum of epipolar constraints over the camera poses in the sliding window. The latter suggests that the Gauss-Newton normal equations scale only with camera poses, as opposed to the larger system sizes we typically obtain with sparse bundle adjustment. For a window of size $n = 2$, the cost function is *scale unaware* and becomes the *essential matrix* least squares formulation [25]. For $n > 2$, at least one camera pose is taken as constant and the optimization problem becomes *scale aware* owed to the presence of measurements in more than two camera views.

IV. EXPERIMENTS

We executed our method on video sequences of the river shore recorded on a cruising vehicle averaging 5-7 knots speed. Each sequence is accompanied by IMU samples at 150 Hz and GPS readings at approximately 1Hz (standard refresh rate of USB dongle). Both IMU and GPS data are time-stamped with a video frame index. The GPS readings are used as our main measure of "ground truth".

A. Ground truth

Due to the limitations imposed primarily by the environment, GPS was the only reliable means of a ground truth estimate. Although the GPS position reading is known to have approximately 1.5 meters variance, it however is highly reliable in long routes, simply because it does not accumulate error. The routes corresponding to the recorded sequences are longer than 30 meters and therefore, in this context, the GPS routes are accurate representations of ground truth.

The scale of the vision-based estimated trajectory is defined by the last reading of speed over ground from the

GPS, before the visual odometer takes over. Since this is a fairly noisy estimate and scale errors accumulate during SLAM, direct comparisons between the GPS trajectory and the respective SLAM-generated estimate would be biased, regardless of whether this estimate is adequate to safely localize the vehicle while cruising. Thus, in order to perform unbiased comparisons, both trajectories are interpolated with a Catmull-Rom spline [26] and parametrized relative to their length; using the common parameter, we estimate the optimal (in the least squares sense) affine transformation that minimizes the distances between the two splines for common parameter values. The choice of interpolating spline can be arbitrary, but Catmull-Rom curves are more convenient to construct locally with only four points at a time. It should be stressed that in order to obtain corresponding parameters values between the GPS and vision based trajectories, it is necessary to re-parametrize the curves by means of arc length. The general concept is illustrated in Figure 8.

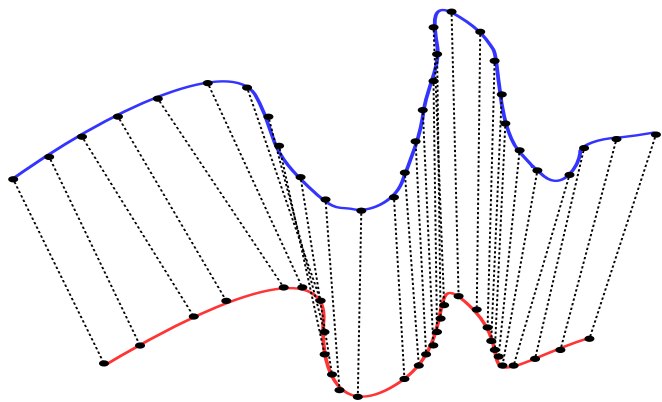


Fig. 8: Parameter correspondence between the GPS (blue) and visual odometry (red) interpolated curves irrespectively of scale. Synchronization is achieved using the GPS-frame index log.

Suppose we fit two splines $g(t_g)$ and $v(t_v)$ where $t_g, t_v \in [0, 1]$ to both GPS and vision based odometry points using a common parameter $u \in [0, 1]$. It is necessary for this parameter to reflect percentage of overall arc-length in either curve. Thus, the warping functions for $t_g(u)$ and $t_v(u)$ can be computed from the following differential equations:

$$s_g(u) = \int_0^{t_g(u)} g(t) dt \quad (15)$$

$$s_v(u) = \int_0^{t_v(u)} v(t) dt \quad (16)$$

where s_g and s_v are the arc-lengths of g and h respectively. Having obtained the reparametrized splines in terms of u , it is now a matter of ordinary least squares to fit a 2D affine transformation A that minimizes the distance between a number of sampled points in g and v for common values

of u :

$$\hat{A} = \underset{A}{\operatorname{argmin}} \left\{ \sum_{k=0}^{n-1} \left\| Av \left(t_v \left(\frac{k}{n-1} \right) \right) - g \left(t_g \left(\frac{k}{n-1} \right) \right) \right\|^2 \right\} \quad (17)$$

where n is the number of uniform samples in $[0,1]$.

B. Odometry estimates

Figures 9 and 10 illustrate odometry estimates and GPS ground truth superimposed on satellite imagery for two indicative sequences. Overall data was recorded in the Tamar river, Devon, UK. The trajectories are generally long (several hundred meters); unfortunately, the vehicle had to follow specific routes in order to avoid shallow waters and this many times resulted in trajectories without many twists. It should be noted however that transient rotational and linear motion is generally rich as shown in the actual footage, yet it is not reflected in the GPS based trajectory for obvious reasons.

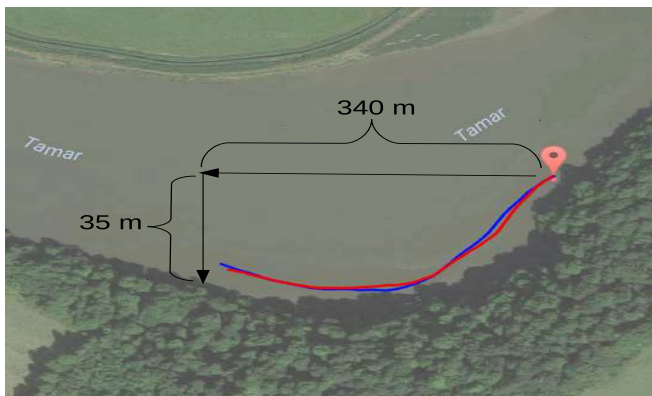


Fig. 9: GPS ground truth spline (blue) and estimated visual odometry spline (red) overlaid in satellite imagery. Approximate distances along the x and y axes are given to indicate scale.

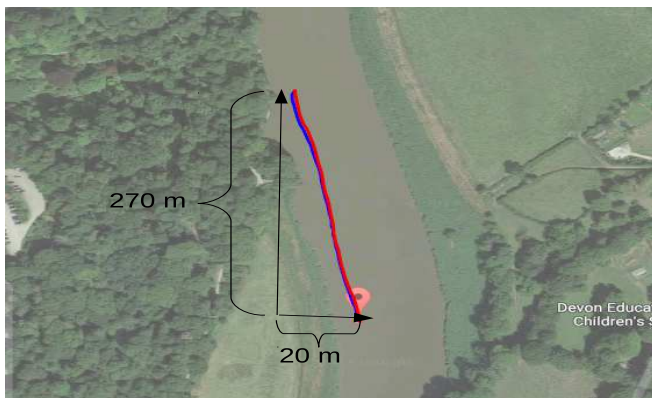


Fig. 10: GPS ground truth spline (blue) and estimated visual odometry spline (red) overlaid in satellite imagery. Approximate distances along the x and y axes are given to indicate scale.

We evaluated the quality of localization in a total of 8 trajectories, averaging approximately 350m total distance

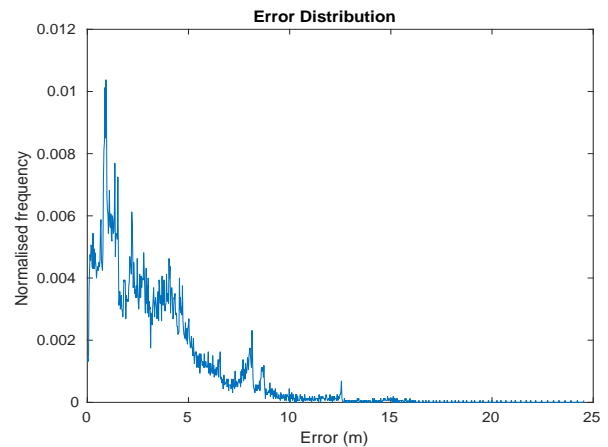


Fig. 11: Position error distribution obtained from 8 routes in the river.

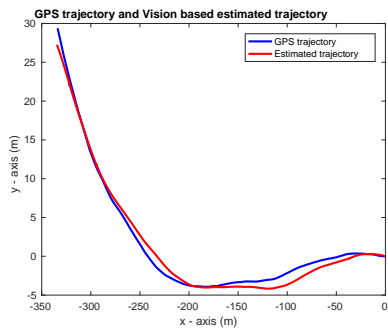
covered on the water. The position error was evaluated in meters using the method described in Section IV-A. On average, the maximum position error is typically in the range of 8-10m with the exception of occasional occurrences in the 20m range, such as in the 400m long trajectory of Figure 15. The position error distribution obtained from all 8 sequences is depicted in Figure 11. Interestingly, the plot is very reminiscent of the χ^2 distribution, which empirically indicates consistency with normally distributed relative pose squared error.

Figures 12, 13, 15, 14 illustrate plots generated from 4 selected sequences, illustrating comparison of visual odometry with ground truth, position error distribution and its progression with distance traveled by the vehicle.

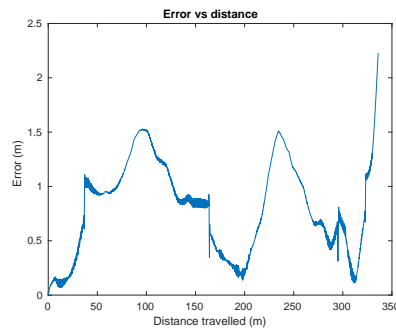
V. CONCLUSION

We have presented a method to obtain reliable visual odometry estimates for an unmanned sea-surface vehicle using imagery of the shore, in order to accommodate autonomous cruise when GPS feedback becomes unavailable. To the best of our knowledge, this is the first time that a sea-surface vehicle location is obtained exclusively with standard Visual SLAM techniques. Except for the typical shortcomings of outdoor visual SLAM (brightness, shading, motion blur, environmental conditions, etc.), the application at hand presents us with additional challenges associated with the absence of a ground plane and extreme scene-depth variation. Typically, the ground plane endows the imagery with features that correspond to near-by world points based on which, camera motion can be reliably estimated. In contrast, for imagery obtained on a cruising boat, there is no ground plane and motion must be estimated from features in the background which may or may not be near, as depth varies significantly.

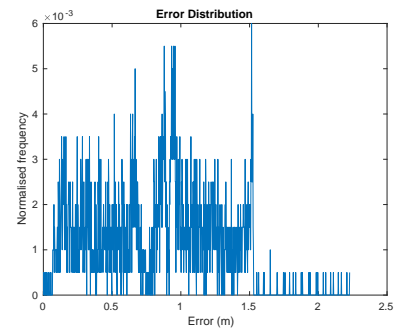
To cope with depth variation in the useful portion of the background, we advocate the use of structure-less visual SLAM with the aid of orientation priors from an IMU. With this information, we are able to cast the relative pose



(a) Visual odometry vs GPS trajectory.

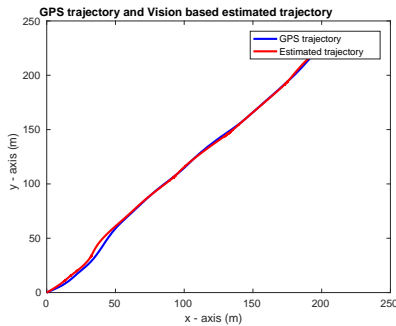


(b) Position error vs Distance travelled.

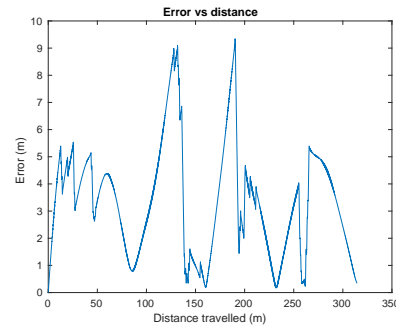


(c) Error distribution.

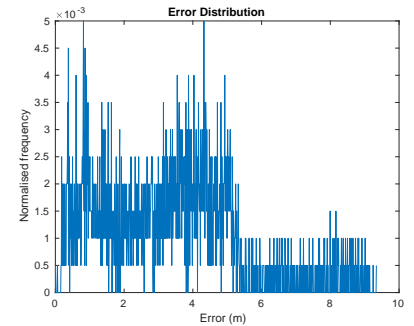
Fig. 12: An approximately 340m long route along the Tamar river, near Halton Quay, Devon, UK.



(a) Visual odometry vs GPS trajectory.

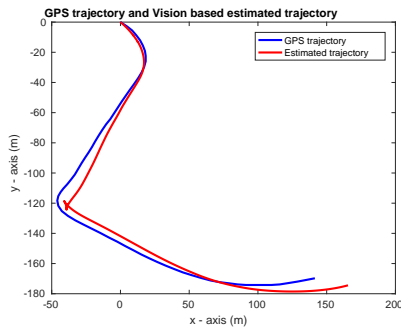


(b) Position error vs Distance travelled.

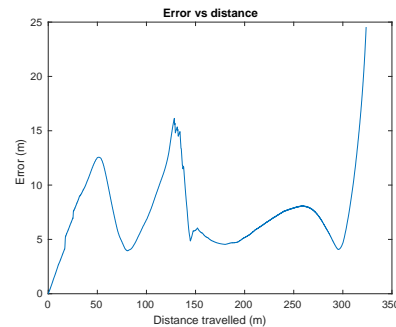


(c) Error distribution.

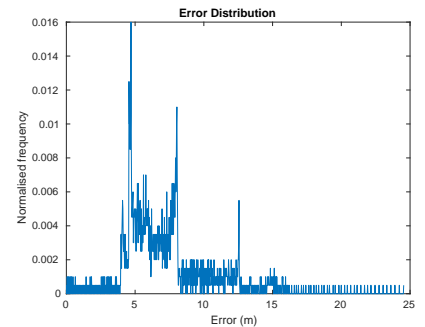
Fig. 13: An approximately 285m long route along the Tamar river, in Bohetherick, Devon, UK.



(a) Visual odometry vs GPS trajectory.

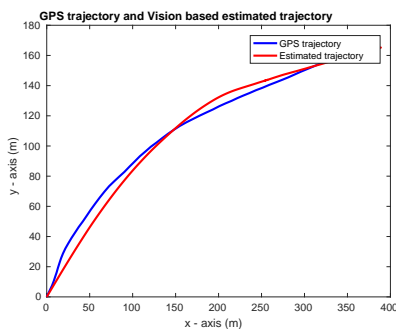


(b) Position error vs Distance travelled.

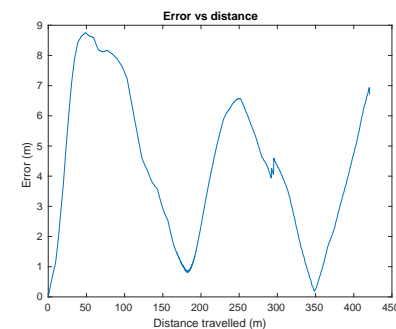


(c) Error distribution.

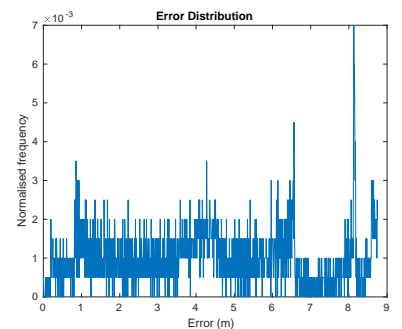
Fig. 14: An approximately 400m long route along the Tamar river, in Calstock, Devon, UK.



(a) Visual odometry vs GPS trajectory.



(b) Position error vs Distance travelled.



(c) Error distribution.

Fig. 15: An approximately 450m long route along the Tamar river, near Cotehele, Devon, UK.

problem directly in terms of image projections, thereby circumventing the map which may contain a number of noisy reconstructions. Our relative pose equations do not discard distant points, but they simply ignore them when their disparity is not large enough to provide additional motion information. However, regardless of depth, a point can always be an outlier, so in order to combine our model with RANSAC, we propose a robust error reflecting epipolar plane misalignment. The error is valid for any type of degenerate configuration, including pure rotational motion. To avoid having to use the map during error (reprojection) refinement, we employ a cost function which is formed by the sum of standard epipolar constraints.

The specifics of the application described herein, naturally do not exactly match the conditions in available datasets. We therefore recorded our own sequences and timestamped inertial data and GPS readings with frame indexes in each sequence. To obtain unbiased comparisons with ground truth, it was necessary to compare the GPS-generated trajectory with visual odometry using a common parameter that reflects equal proportions of arc-length with respect to the origin in both curves. To do so, we interpolate both curves using a spline and thereafter, reparametrize resulting curves in terms of their arc-length in the interval $[0, 1]$. The use of a common arc-length parameter allows for actual-scale comparisons with the GPS trajectory.

REFERENCES

- [1] O. Dunkley, J. Engel, J. Sturm, and D. Cremers, "Visual-inertial navigation for a camera-equipped 25g nano-quadrotor," in *IROS2014 aerial open source robotics workshop*, 2014, p. 2.
- [2] D. Schleicher, L. M. Bergasa, M. Ocaña, R. Barea, and M. E. López, "Real-time hierarchical outdoor slam based on stereovision and gps fusion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 3, pp. 440–452, 2009.
- [3] K. Konolige, M. Agrawal, and J. Sola, "Large-scale visual odometry for rough terrain," in *Robotics research*. Springer, 2010, pp. 201–212.
- [4] M. Faessler, F. Fontana, C. Forster, E. Mueggler, M. Pizzoli, and D. Scaramuzza, "Autonomous, vision-based flight and live dense 3d mapping with a quadrotor micro aerial vehicle." *J. Field Robotics*, vol. 33, no. 4, pp. 431–450, 2016.
- [5] S. Griffith, F. Dellaert, and C. Pradalier, "Robot-enabled lakeshore monitoring using visual slam and sift flow," in *RSS Workshop on Multi-View Geometry in Robotics*. Citeseer, 2015.
- [6] C. Harris and M. Stephens, "A combined corner and edge detector." in *Alvey vision conference*, vol. 15, no. 50. Citeseer, 1988, pp. 10–5244.
- [7] B. D. Lucas, T. Kanade, *et al.*, "An iterative image registration technique with an application to stereo vision," 1981.
- [8] J.-Y. Bouguet, "Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm," *Intel Corporation*, vol. 5, no. 1-10, p. 4, 2001.
- [9] W. Naeem, T. Xu, R. Sutton, and J. Chudley, "Design of an unmanned surface vehicle for environmental monitoring," 2006.
- [10] G. Klein and D. Murray, "Parallel tracking and mapping on a camera phone," in *Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on*. IEEE, 2009, pp. 83–86.
- [11] P. H. Torr, A. Zisserman, and S. J. Maybank, "Robust detection of degenerate configurations for the fundamental matrix," in *Computer Vision, 1995. Proceedings., Fifth International Conference on*. IEEE, 1995, pp. 1037–1042.
- [12] K. Kanatani, "Geometric information criterion for model selection," *International Journal of Computer Vision*, vol. 26, no. 3, pp. 171–189, 1998.
- [13] P. H. Torr, A. Zisserman, and S. J. Maybank, "Robust detection of degenerate configurations while estimating the fundamental matrix," *Computer Vision and Image Understanding*, vol. 71, no. 3, pp. 312–333, 1998.
- [14] R. I. Hartley, "Ambiguous configurations for 3-view projective reconstruction," in *European Conference on Computer Vision*. Springer, 2000, pp. 922–935.
- [15] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual-inertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.
- [16] M. Lourakis and E. Hourdakis, "Planetary rover absolute localization by combining visual odometry with orbital image measurements," in *Proc. of the 13th Symposium on Advanced Space Technologies in Automation and Robotics (ASTRA15)*. European Space Agency, 2015.
- [17] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 1. Ieee, 1999, pp. 666–673.
- [18] —, "A flexible new technique for camera calibration," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.
- [19] G. Bradski *et al.*, "The opencv library," *Doctor Dobbs Journal*, vol. 25, no. 11, pp. 120–126, 2000.
- [20] H. Zhuang and Z. S. Roth, *Camera-aided robot calibration*. CRC press, 1996.
- [21] L. Kneip, M. Chli, R. Siegwart, *et al.*, "Robust real-time visual odometry with a single camera and an imu," in *BMVC*, 2011, pp. 1–11.
- [22] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [23] P. H. Torr and A. Zisserman, "Mlesac: A new robust estimator with application to estimating image geometry," *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 138–156, 2000.
- [24] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [25] G. Terzakis, "Relative camera pose recovery and scene reconstruction with the essential matrix in a nutshell," Technical report MIDAS.SNMSE.2013.TR.007, Marine and Industrial Dynamic Analysis School of Marine Science and Engineering, Plymouth University, Tech. Rep., 2013.
- [26] E. Catmull and R. Rom, "A class of local interpolating splines," *Computer aided geometric design*, vol. 74, pp. 317–326, 1974.