2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Pipe Mapping with Monocular Fisheye Imagery

Peter Hansen, Hatem Alismail, Peter Rander and Brett Browning

Abstract— **We present a vision-based mapping and localization system for operations in pipes such as those found in Liquified Natural Gas (LNG) production. A forward facing fisheye camera mounted on a prototype robot collects imagery as it is tele-operated through a pipe network. The images are processed offline to estimate camera pose and sparse scene structure where the results can be used to generate 3D renderings of the pipe surface. The method extends state of the art visual odometry and mapping for fisheye systems to incorporate geometric constraints based on prior knowledge of the pipe components into a Sparse Bundle Adjustment framework. These constraints significantly reduce inaccuracies resulting from the limited spatial resolution of the fisheye imagery, limited image texture, and visual aliasing. Preliminary results are presented for datasets collected in our fiberglass pipe network which demonstrate the validity of the approach.**

I. INTRODUCTION

Pipe inspection is a critical task to a number of industries, including Natural Gas production where pipe surface structure changes at the scale of millimeters are of concern. In this work, we report on the development of a fisheye visual odometry and mapping system for an in-pipe inspection robot (e.g. [1]) to produce detailed, millimeter resolution 3D surface structure and appearance maps. By registering maps over time, changes in structure and appearance can be identified, which are both cues for corrosion detection. Moreover, these maps can be imported into rendering engines for effective visualization or measurement and analysis.

In prior work [2], we developed a verged perspective stereo system capable of measuring accurate camera pose and producing dense sub-millimeter resolution maps. A limitation of the system was the inability of the camera to image the entire inner surface of the pipe. Here, we address this issue by using a forward-facing wide-angle fisheye camera mounted on a small robot platform, as shown in Fig. 1a. This configuration enables the entire inner circumference to be imaged from which full coverage appearance maps can be produced. Fig. 1b shows the constructed 400mm (16 inch) internal diameter pipe network used in the experiments. This pipe diameter is commonly used in Liquified Natural Gas (LNG) processing facilities, which is a target domain for our system. Sample images from the fisheye camera are shown in Figs. 1c and 1d. The extreme lighting variations evident in the sample images pose significant challenges during image processing, as discussed in Section III.

This publication was made possible by NPRP grant #08-589-2-245 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

Hansen is with the QRI8 lab, Carnegie Mellon University, Doha, Qatar phansen@qatar.cmu.edu. Alismail, Rander, and Browning are with the Robotics Institute/NREC, Carnegie Mellon University, Pittsburgh PA, USA, {halismai,rander,brettb}@cs.cmu.edu

(a) Robot. (b) Test pipe network.

(c) Image in straight section. (d) Image in T-intersection.

Fig. 1: The prototype robot with forward facing fisheye camera (a), 400mm (16 inch) internal diameter pipe network (b), and sample images logged in a straight section (c) and T-intersection (d) during tele-operation through the network. Lighting is provided by 8 LEDs surrounding the camera.

Our system builds from established visual odometry and multiple view techniques for central projection cameras [3], [4], [5], [6]. Binary thresholding, morphology and shape statistics are first used to classify straight sections and T-intersection. Pose and structure results are obtained for each straight section using a sliding window Sparse Bundle Adjustment (SBA) and localized straight cylinder fitting/regularization within the window. Fitting a new straight cylinder each window allows some degree of gradual pipe curvature to be modeled, e.g. sag in the pipes. After processing each straight section, results for the T-intersections are obtained using SBA and a two cylinder intersection fitting/regularization – the two cylinders are the appropriate straight sections of the pipe network. When applicable, loop closure with g^2o [7] is implemented using pose estimates from visual correspondences in overlapping sections of a dataset. As a final step, the pose and structure estimates are used to produce a dense point cloud rendering of the interior surface of the pipe network.

Results are presented in Section IV which show the visual odometry and sparse scene reconstruction for two datasets collected in our pipe network. Dense 3D point cloud renderings for one dataset are also provided. These preliminary results illustrate the validity of the proposed system.

II. FISHEYE CAMERA

We use a 190° angle of view Fujinon fisheye lens fitted to a $1280pix \times 960pix$ resolution CCD firewire camera. Image formation is modeled using a central projection polynomial mapping, which is a common selection for fisheye cameras (e.g. [8], [9]). A scene point $\mathbf{X}_i$ projects to a coordinate $\boldsymbol{\eta}(\theta, \phi) = \mathbf{X}_i / ||\mathbf{X}_i||$ on the camera's unit view sphere centered at the single effective viewpoint $(0,0,0)^T$. The angles θ and ϕ are, respectively, colatitude and longitude. The projected fisheye image coordinates $\mathbf{u}(u, v)$ are

$$\mathbf{u} = \begin{bmatrix} (k_1\theta + k_2\theta^3 + k_3\theta^4 + k_4\theta^5)\cos\phi + u_0 \\ (k_1\theta + k_2\theta^3 + k_3\theta^4 + k_4\theta^5)\sin\phi + v_0 \end{bmatrix}, \quad (1)$$

where $\mathbf{u_0}(u_0, v_0)$ is the principal point. Multiple images of a checkerboard calibration target with known Euclidean geometry were collected, and the model parameters fitted using a non-linear minimization of the sum of squared checkerboard grid point image reprojection errors.

III. VISUAL ODOMETRY AND MAPPING

The visual odometry (VO) and mapping procedure is briefly summarized as follows:

A. Perform feature matching/tracking with keyframing.
B. Divide images into straight sections and T-intersections.
C. Obtain VO/structure estimates for each straight section using a sliding window SBA and localized straight cylinder fitting/regularization.
D. Obtain VO/structure estimates for each T-intersection using a two cylinder T-intersection model. This step effectively merges the appropriate straight sections.
E. Perform loop closure when applicable.

The visual odometry steps (C and D) use different cylinder fitting constraints to obtain scene structure errors included as a regularization error in SBA. As previously mentioned, we have observed this to be a critically important step which significantly improves the robustness and accuracy of the visual odometry and scene reconstruction estimates in the presence of: limited spatial resolution from the fisheye camera; feature location noise due to limited image texture and extreme lighting variations; and an often high percentage of feature tracking outliers due again to limited image texture and visual aliasing. At present an average *a priori* metric measurement of the pipe radius r is used during cylinder fitting. Cylinder fitting with a supplied pipe radius also resolves monocular visual odometry scale ambiguity.

A. Feature Tracking

An efficient region-based Harris detector [10] based on the implementation in [6] is used to find a uniform feature distribution in each image. The image is divided into 2×2 regions, and the strongest $N = 200$ features per region are retained. Initial temporal correspondences between two images, image i and image j, are found using cosine similarity matching of 11×11 grayscale template descriptors for each feature. Each of these 11×11 template descriptors is interpolated from a 31×31 region surrounding the feature. Five-point relative pose [3] and RANSAC [11] are used to remove outliers and provide an initial estimate of the essential matrix E. We experimented with multiple scale-invariant feature detectors/descriptors (e.g. SIFT [12], SURF [13]), but observed no significant improvements in matching performance.

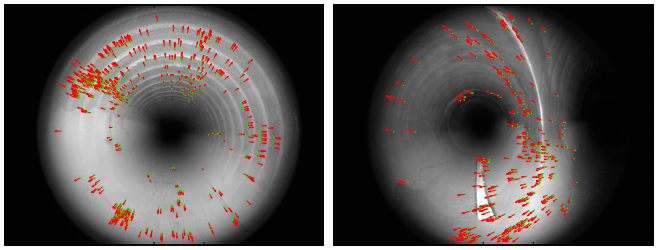


Fig. 2: Sparse optical flow vectors in a straight section (left) and T-intersection (right) obtained using a combination of Harris feature matching and epipolar guided ZNCC.

For all unmatched features in image i, a guided Zero-mean Normalized Cross Correlation (ZNCC) is applied to find their pixel coordinate in image j. Here, guided refers to a search within an epipolar *region* in image j. Since we implement ZNCC in the original fisheye imagery, we back project each integer pixel coordinate to a spherical coordinate $\boldsymbol{\eta}$, and constrain the epipolar search regions using $|\boldsymbol{\eta}_j^T \, E \, \boldsymbol{\eta}_i| < thresh$ — the subscripts denote image i and j. As a final step we implement image keyframing, selecting only images separated by a minimum median sparse optical flow magnitude or minimum percentage correspondences. Both minimums are selected empirically.

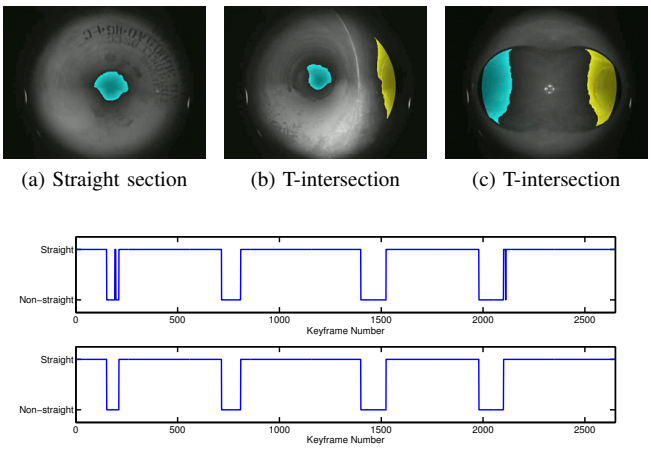
Fig. 2 shows examples of the sparse optical flow vectors for the feature correspondences found between keyframes in a straight section, and keyframes in a T-intersection. Features are 'tracked' across multiple frames by recursively matching using the method described.

The grayscale intensity of an imaged region of the pipe surface can change significantly between frames. This is due primarily to the non-uniform ambient lighting provided by the robot. The cosine similarity metric for matching template descriptors and ZNCC were both selected to provide some invariance to these intensity changes.

B. Image classification: straight vs. T-intersection

The image keyframes must be divided into straight sections and T-intersection before implementing pose estimating and mapping. To classify each image as belonging to a straight section or T-intersection, the image resolution is first reduced by sub-sampling pixels from every second row and column. A binary thresholding is applied to extract dark blobs within the cameras field of view, followed by binary erosion and clustering of the blobs. The largest blob is selected and the second moments of area L and L_p computed about the blob centroid and principal point, respectively. An image is classified as straight if the ratio L_p/L is less than an empirical threshold; we expect to see a large round blob near the center of images in straight sections.

Figs. 3a through 3c show the blobs extracted in three sample images, and initial classification of each image. After



(a) Straight section (b) T-intersection (c) T-intersection

(d) Initial classification (top), and after temporal filtering (bottom). Each of the 4 T-intersection clusters is a unique T-intersection in the pipe network.

Fig. 3: Straight section and T-intersection image classification. Example initial classifications (a-c), and the classification of all keyframes before and after temporal filtering (d).

initial classification, a temporal filtering is used to correct mis-classification, as illustrated in Fig. 3d. This filtering enforces a minimum straight/T-intersection cluster size.

C. Straight VO: Sliding Window SBA / local straight cylinder

For each new keyframe, the feature correspondences are used to estimate the new camera pose and scene points coordinates. This includes using Nister's 5-point algorithm to obtain an initial unit-magnitude pose change estimate, optimal triangulation to reconstruct the scene points [4], and prior reconstructed scene coordinates in the straight section to resolve relative scale. After every 50 keyframes, a modified sliding window SBA is implemented which includes a localized straight cylinder fitting used to compute a scene point regularization error. A 100 keyframe window size is used which, for a typical dataset, equates to a segment of pipe approximately one meter in length. This SBA is a multi-objective least squares minimization of image reprojection errors $\epsilon_{\mathbf{I}}$ and scene point errors $\epsilon_{\mathbf{X}}$. An optimal estimate of the camera poses P and scene points $\mathbf{X}$ in the window, as well as the fitted cylinder C are found which minimize the combined error ϵ:

$$\epsilon = \epsilon_{\mathbf{I}} + \tau\, \epsilon_{\mathbf{X}}. \tag{2}$$

The parameter τ is a scalar weighting which controls the trade-off between the competing error terms $\epsilon_{\mathbf{I}}$ and $\epsilon_{\mathbf{X}}$.

The image reprojection error $\epsilon_{\mathbf{I}}$ is the sum of squared differences between all valid feature observations $\mathbf{u}$ and reprojected scene point coordinates $\mathbf{u}'$:

$$\epsilon_{\mathbf{I}} = \sum_i ||\mathbf{u}_i - \mathbf{u}'_i||^2, \tag{3}$$

where $\mathbf{u}(u, v)$ and $\mathbf{u}'(u', v')$ are both inhomogeneous fisheye image coordinates.

The scene point error term $\epsilon_{\mathbf{X}}$ is the sum of squared errors between the optimized scene point coordinates $\mathbf{X}$ and a fitted straight cylinder. The cylinder pose C is defined relative to the first camera pose $P_m = [R_m|\mathbf{t}_m]$ in the sliding window as the origin. It is parameterized using 4 degrees of freedom:

$$\begin{aligned} C &= [\tilde{R}\,|\,\tilde{\mathbf{t}}] \\ &= [R_X(\gamma)\, R_Y(\beta)\,|\,(t_X, t_Y, 0)^T], \end{aligned} \tag{4}$$

where R_A denotes a rotation about the axis A, and t_A denotes a translation in the axis A. Each scene point coordinate $\mathbf{X}_i$ maps to a coordinate $\tilde{\mathbf{X}}_i$ in the cylinder frame using

$$\tilde{\mathbf{X}}_i = \tilde{R}\,(R_m \mathbf{X}_i + \mathbf{t}_m) + \tilde{\mathbf{t}}. \tag{5}$$

The regularization error $\epsilon_{\mathbf{X}}$ is

$$\epsilon_{\mathbf{X}} = \sum_i \left(\sqrt{\tilde{X}_i^2 + \tilde{Y}_i^2} - r \right)^2, \tag{6}$$

where the pipe radius r is supplied. Referring to (2), we use an empirically selected value $\tau = 2.5 \times 10^5$.

As noted previously, there are frequently many feature correspondence outliers resulting from the challenging raw imagery. To minimize the influence of outliers, a Huber weighting is applied to individual error terms before computing $\epsilon_{\mathbf{I}}$ and $\epsilon_{\mathbf{X}}$. Outliers are also removed at multiple stages (iteration steps) using Median Absolute Deviation of the set of all signed Huber weighted errors $\mathbf{u} - \mathbf{u}'$. This outlier removal stage is beneficial when the percentage of outliers is large.

D. T-intersections

The general procedure for processing the T-intersections is illustrated in Fig. 4. After processing each straight section, straight cylinders are fitted to the scene points in the first and last 1 meter segment (Fig. 4a). In both cases, these cylinders are fitted with respect to the first and last camera poses as the origins, respectively, using the parameterization in (4).

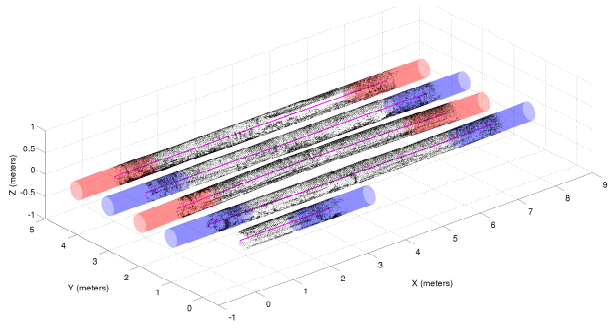
As illustrated in Fig. 4b, a T-intersection is modeled as two intersecting straight cylinders; the red cylinder axis intersects the blue cylinder axis at a unique point $\mathcal{I}$. Let P_r be the first/last camera pose in a red section, and C_r be the cylinder fitted with respect to this camera as the origin. Similarly, let P_b be the last/first camera pose in a blue section, and C_b be the cylinder fitted with respect to this camera as the origin. The parameters ζ_r and ζ_b are rotations about the axis of the red and blue cylinders, and l_r and l_b are the signed distances of the cylinder origins $\mathcal{O}(C_r)$ and $\mathcal{O}(C_b)$ from the intersection point $\mathcal{I}$. Finally, ϕ is the angle of intersection between the two cylinder axes in the range $0° \leq \phi < 180°$. These parameters fully define the change in pose Q between P_b and P_r, and ensure that the two cylinder axes intersect at a single unique point $\mathcal{I}$. Letting

$$D = p\left([R_Z(\zeta_r)|(0,0,l_r)^T], [R_Z(\zeta_b)\, R_Y(\phi)|(0,0,l_b)^T]\right), \tag{7}$$

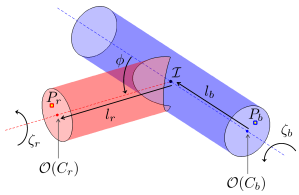
where $p(b, a)$ is a projection a followed by b, and R_A is a rotation about axis A, then

$$Q = p\left(inv(C_r), p(D, C_b)\right), \tag{8}$$

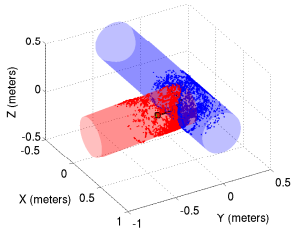
where $inv(C_r)$ is the inverse projection of C_r.



(a) Straight sections with cylinders fitted to endpoints.



(b) The two cylinder T-intersection model parameters (refer to text for detailed description). The red and blue cylinders have the same internal radius r and are constrained to intersect at a unique point $\mathcal{I}$.



(c) Visual odometry and scene reconstruction result using the T-intersection model. The scene points have been automatically assigned to each cylinder allowing cylinder fit regularization terms to be evaluated.

Fig. 4: A T-intersection is modeled as the intersection of cylinders fitted to the straight sections of the pipe. Respectively, the blue and red colors distinguish the horizontal and vertical sections of the 'T', as illustrated in (b).

SBA is used to optimize all camera poses P_T in the T-intersection between P_r and P_b, as well as all new scene points $\mathbf{X}$ in the T-intersection, and the T-intersection model parameters $\Phi(\zeta_r, l_r, \zeta_b, l_b, \phi)$. Again, the objective function minimized is the same form as (2), which includes an image reprojection error (3) and scene fitting error (6). The same value $\tau = 2.5 \times 10^5$, robust cost function, and outlier removal scheme are also used.

Special care needs to be taken when computing the scene fitting error $\epsilon_{\mathbf{X}}$ in (6) as there are two cylinders C_r, C_b in the T-intersection model. Specifically, we need to assign each scene point to one of the cylinders, and compute the individual error terms in (6) with respect to this cylinder. This cylinder assignment is performed by finding the distance to each of the cylinder surfaces, and selecting the cylinder for which the absolute distance is a minimum. Fig. 4c shows the results for one of the T-intersections after SBA has converged. The color-coding of the scene points (dots) represent their cylinder assignment.

E. Loop Closure

For multi-loop datasets, loop closure is implemented using the graph based optimization algorithm g^{2}o [7]. The graph vertices are the set of all camera poses described by their Euclidean coordinates and orientations (quaternions). The graph edges connect temporally adjacent camera poses, and the loop closures connecting camera poses in the same section of the pipe network visited at different times. Each edge has an associated relative pose estimate between the vertices it connects, and a covariance of the estimate. Alternate graph-based loop closure techniques with comparable performance and improved efficiency could also be used (e.g. [14]).

Fig. 5: Pipe network: T1 through T4 are the T-intersections.

Our constrained environment enables potential loop closures to be selected without the need for visual place recognition techniques, such as those based on bag of visual words (BoW)[15]. This is achieved by knowing the section of the network where the robot begins, and simply counting left/right turns – see Fig. 5. Let $P_\mathbf{i}$ and $P_\mathbf{j}$ be the poses in the same straight section of pipe visited at different times. We compute the Euclidean distances $l_\mathbf{i}$ and $l_\mathbf{j}$ to the T-intersection centroid $\mathcal{I}$ at the end of the straight section. Poses P_i are P_j are selected as a candidate pair if $l_i \approx l_j$.

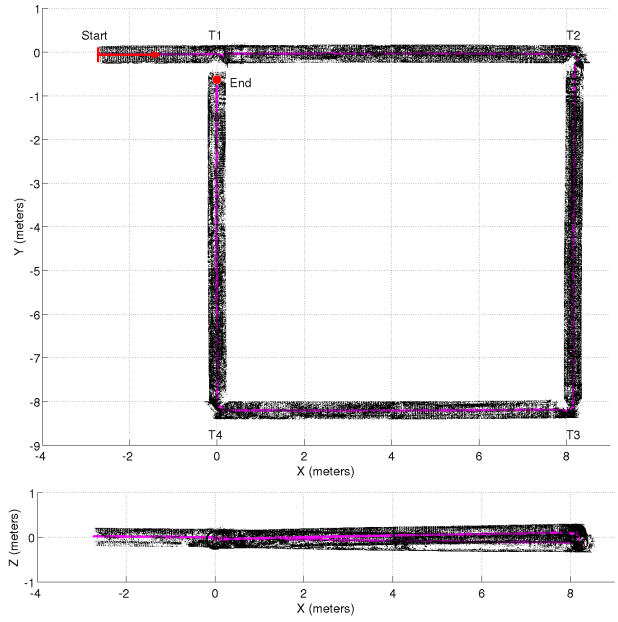
Once candidate loop closure poses $P_i \leftrightarrow P_j$ are selected, the relative pose estimate is refined using image correspondences found by matching SIFT descriptors for the previously detected Harris corners. SIFT descriptors are used at this stage to achieve improved rotation invariance. Prior knowledge of the surrounding scene structure (fitted cylinders) is used to resolve the monocular scale ambiguity of the relative pose estimates obtained using the five-point algorithm and RANSAC, followed by non-linear refinement.

IV. EXPERIMENTS AND RESULTS

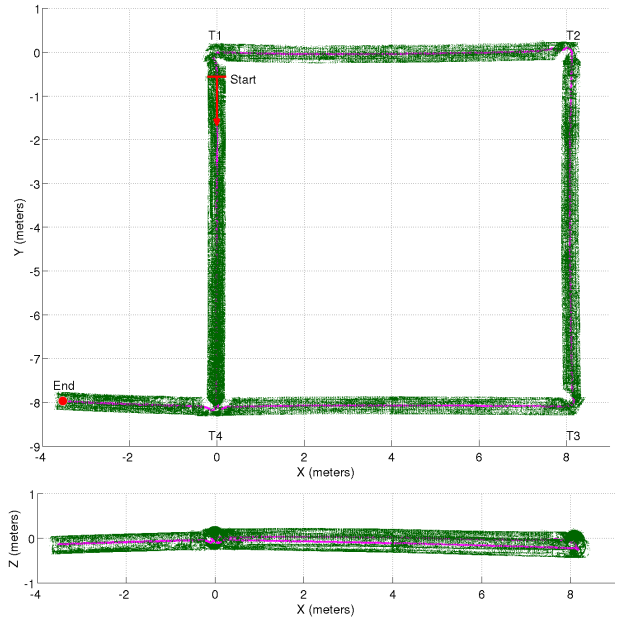
Two grayscale monocular fisheye image datasets ($1280pix \times 960pix$ resolution, 7.5 frames per second) were collected in our constructed 400mm (16 inch) internal diameter fiberglass pipe network shown in Fig. 5. For this, the robot was tele-operated using images streamed over a wireless link, and all lighting was provided by 8 high intensity LEDs equipped on the robot — see Fig. 1. Dataset A is a near full loop of the pipe network containing approximately 26,000 images, from which 2760 keyframes were automatically selected. Dataset B is a full loop datasets containing approximately 24,000 images and 4170 keyframes. All image processing steps described in Section III were implemented offline.

A. Visual Odometry

The visual odometry and sparse scene reconstruction results for each straight section in dataset A were shown previously in Fig. 4a. The complete results for both datasets are provided in Fig. 6. The labels T1 through T4 correspond to those in Fig. 5. Note that no loop closure was possible for dataset A. For dataset B, loop closure was implemented using 15 loop closure poses in the straight section between the T-intersections T1 and T4.



(a) Dataset A: Robot moved in the sequence: start-T1-T2-T3-T4-end.



(b) Dataset B: Robot moved in the sequence: start-T4-T3-T2-T1-T4-end.

Fig. 6: Visual odometry (magenta line) and sparse scene reconstructions for dataset A (no loop closure) and dataset B (loop closure between T1 and T4).

An ideal performance metric for our system is direct comparison of the visual odometry (camera pose) and scene structure results to accurate ground truth. However, obtaining suitable ground truth is particularly challenging due to the unavailability of GPS in a pipe, and limited accuracy of standard grade Inertial Measurement Units (IMUs).

Our current *approximate* ground truth is hand-held laser distance measurements estimating the distance between all T-intersection centroids $\mathcal{I}'$ (see Fig. 4c). These measurements are compared to the visual odometry T-intersections centroid

Distance	T1-T2	T2-T3	T3-T4	T4-T1	T1-T3	T2-T4
Laser (m)	8.150	8.174	8.159	8.110	11.468	11.493
A: VO (m)	8.184	8.131	8.138	8.161	11.514	11.543
A: Error (m)	0.034	-0.043	-0.021	0.051	0.046	0.050
A: Error (%)	0.42	-0.53	-0.26	0.63	0.41	0.44
B: VO (m)	8.114	8.118	8.142	8.105	11.473	11.492
B: Error (m)	-0.036	-0.056	-0.017	-0.005	0.005	-0.001
B: Error (%)	-0.44	-0.69	-0.21	-0.07	0.04	-0.01

TABLE I: The T-intersection center-to-center distances obtain with a hand-held laser (approximate ground truth – see text in Section IV-A for detailed description), and visual odometry for dataset A (without loop closure) and dataset B (with loop closure) – refer to Fig. 6. The signed error percentages are given with respect to the laser measurements.

$\mathcal{I}$ distances in Table I. The largest absolute error measured is 0.056m (0.69%) between T3-T4 for dataset B.

The laser ground truth measurements are only approximations as, for practical reasons, the measurements were taken between the centers of the upper exterior surfaces of each T-intersection. These reference points do not correlate directly to the 'true' centroids $\mathcal{I}$. Moreover, there were minor alterations to the pipe network in the period between collecting the datasets (sections of pipe temporarily removed then placed back). The errors reported in table I may be within the bounds of the laser ground truth estimate uncertainty.

In practice, modeling each long straight section of our pipe network as a perfect straight cylinder is too restrictive. Firstly, each individual pipe segment contains some degree of curvature/sag. Secondly, the individual segments used to construct the long straight sections of pipe (see Fig. 5) are not precisely aligned. It is for this reason that we only perform straight cylinder fitting locally within the 100 keyframe sliding window SBA, which typically spans a 1 meter length of pipe, and limit the maximum value for τ in (2). Doing so permits some gradual pipe curvature to be achieved in the results, as evident in Fig. 4a. However, for severe pipe misalignments, deformations, or elbow joints, we expect that the accuracy of results would rapidly deteriorate. A cubic spline modeling of the cylinder axis may be required in these scenarios, despite the significant increase in computational expense when computing the scene point regularization errors. We aim to address this issue in future work.

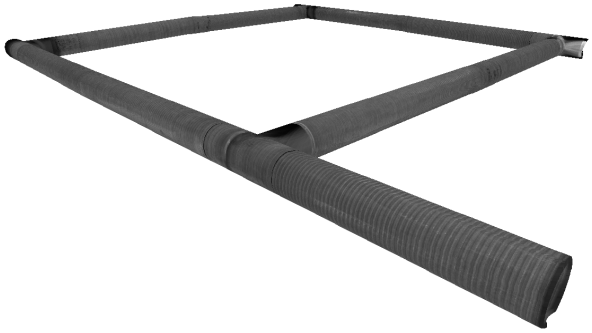
B. Dense Rendering

Using the visual odometry results for dataset A, an appearance map of the pipe network was produced which may be used as input for automated corrosion algorithms and direct visualization in rendering engines. Fig. 7 shows the appearance map of the pipe network, including a zoomed in view of a small straight section (Fig. 7a) and T-intersection (Fig. 7b) to highlight the detail. The consistency of the appearance, namely the lettering on the pipes, demonstrates accurate visual odometry estimates.

The appearance map produced is a dense 3D grayscale point cloud which could be extended to a full facet model. The Euclidean point cloud coordinates were set using the

(a) Small straight section 1 meter in length. Middle column shows an original fisheye image (top), and dense reconstruction near the same location in the pipe (bottom). Right column is a small section of pipe cropped from original image (top), and dense reconstruction (bottom).

(b) T-intersection.



(c) Full dataset.

Fig. 7: Dense 3D grayscale appearance map of the pipe network. (a) and (b) are zoomed in sections of (c).

cylinder fitting results for both the straight sections and T-intersections. The grayscale intensity value for each point was obtained by mapping it into all valid fisheye image keyframes, and then taking the average sampled image intensity value over all the keyframes. Here, valid is defined as having a projected angle of colatitude $50^\circ < \theta < 90^\circ$ (i.e. near the periphery of the fisheye images where spatial resolution is a maximum). A range of improvements are being developed to mitigate the strong lighting variations in the rendered model. As evident in Fig. 7, these are most prevalent in the T-intersections were the raw imagery frequently contains strong specular reflections and saturation.

V. CONCLUSIONS

An initial fisheye visual odometry and mapping system was presented for non-destructive automated corrosion detection in LNG pipes. To improve the accuracy of the system, various cylinder fitting constraints for straight sections and T-intersections were incorporated as regularization terms in sparse bundle adjustment frameworks. The camera pose estimates and fitted cylinders are used as the basis for constructing dense pipe renderings which may be used for direct visualization.

Results were presented for two datasets logged in a 400mm (16 inch) internal diameter fiberglass pipe network. To evaluate the accuracy of the pipe network reconstruction, we compared the distances between all T-intersections in the network. All distance measurements obtained were well within $\pm 1\%$ of the ground truth laser distance measurements. Moreover, the dense appearance maps produced further highlighted the consistency of the camera pose and scene reconstruction results.

Directions for future work include the development of structured lighting for estimating the internal pipe radius, and spline-based cylinder axis modeling. Both have the potential to improve the accuracy and flexibility of the system.

REFERENCES

[1] H. Schempf, "Visual and NDE inspection of live gas mains using a robotic explorer," *JFR*, vol. Winter, pp. 1–33, 2009.

[2] P. Hansen, H. Alismail, B. Browning, and P. Rander, "Stereo visual odometry for pipe mapping," in *IROS*, 2011, pp. 4020–4025.

[3] D. Nistér, "An efficient solution to the five-point relative pose problem," *PAMI*, vol. 26, no. 6, pp. 756–770, June 2004.

[4] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge Univ. Press, 2003.

[5] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment - a modern synthesis," in *Proceedings of the International Workshop on Vision Algorithms: Theory and Practice*, ser. ICCV '99. London, UK: Springer-Verlag, 2000, pp. 298–372.

[6] D. Nistér, O. Naroditsky, and J. Bergen, "Visual odometry for ground vehicle applications," *Journal of Field Robotics*, vol. 23, no. 1, pp. 3–20, January 2006.

[7] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g^2o: A general framework for graph optimization," in *International Conference on Robotics and Automation*, 2011, pp. 3607–3613.

[8] Y. Xiong and K. Turkowski, "Creating image-based VR using a self-calibrating fisheye lens," in *CVPR*, 1997, pp. 237–243.

[9] J. Kannala and S. Brandt, "A generic camera model and calibration method for conventional, wide-angle and fish-eye lenses," *PAMI*, vol. 28, no. 8, pp. 1335–1340, August 2006.

[10] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings Fourth Alvey Vision Conference*, 1988, pp. 147–151.

[11] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Comms. of the ACM*, pp. 381–395, 1981.

[12] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[13] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, June 2008.

[14] G. Dubbelman, P. Hansen, B. Browning, and M. B. Dias, "Orientation only loop-closing with closed-form trajectory bending," in *International Conference on Robotics and Automation*, 2012.

[15] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *International Conference on Computer Vision*, October 2003, pp. 1470–1477.