Distortion-Resistant Spherical Visual Odometry for UAV-Based Bridge Inspection

Sarthak Pathak[a], Alessandro Moro[a], Hiromitsu Fujii[b], Atsushi Yamashita[a], Hajime Asama[a]

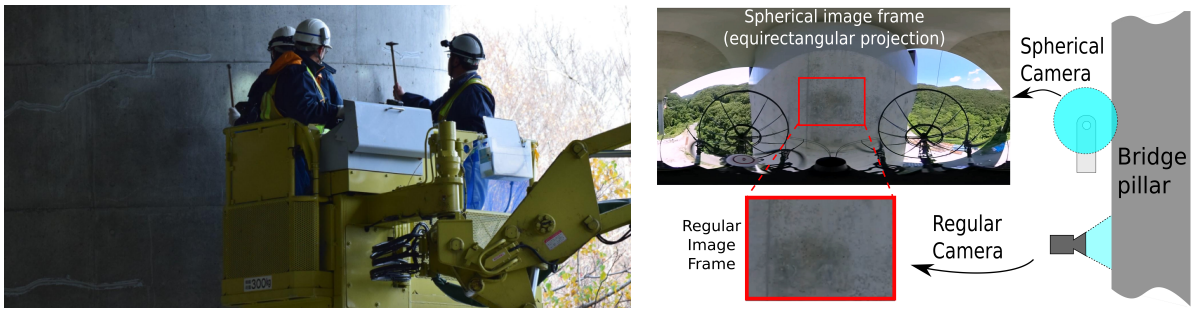
[a]The University of Tokyo, Tokyo, Japan;
[b]Chiba Institute of Technology, Chiba, Japan

ABSTRACT

In this research, we propose a novel distortion-resistant visual odometry technique using a spherical camera, in order to provide localization for a UAV-based, bridge inspection support system. We take into account the distortion of the pixels during the calculation of the 2-frame essential matrix via feature-point correspondences. Then, we triangulate 3D points and use them for 3D registration of further frames in the sequence via a modified spherical error function. Via experiments conducted on a real bridge pillar, we demonstrate that the proposed approach greatly increases the accuracy of localization, resulting in an 8.6 times lower localization error.

Keywords: Spherical vision, Distortion, Visual odometry

1. INTRODUCTION



(a) (b)

Figure 1. (a) Manual close-up bridge inspection: Slow and tedious (b) FoV comparison: A regular camera can only see a small area, so close-up. A spherical camera can view much more.

Tall bridges support large volumes of traffic which induce cyclic loads on them. A single crack or defect could widen and cause catastrophic failure. Hence, they require periodic, close-up inspection. Current inspection technologies are quite tedious as they involve cranes and large hydraulic arms that move people close to the surface. An example of this is shown in Fig. 1 (a).

In order to solve this issue, many inspection methods based on the use of UAVs have been suggested in.[1–3] Equipped with a high-resolution camera and/or other sensors, they can fly close to the structures and map the surface data in order to 'digitize' it for easier, offline inspection.[4,5] For such purposes, there is a need to estimate the 3 dimensional position and orientation of the UAV on the physical structure, in order to map the data collected. GPS technology is insufficient to provide an accurate estimate of the 3D position and orientation. Hence, camera based methods are preferred. There are many approaches that can perform 3D mapping and localization using perspective cameras.[6–8] However, if a normal perspective camera is used in such cases, it will not be able to view more than a tiny section of the structure. Pixels and other distinguishing features would easily flow out of view. Hence, a spherical camera which can see the entire structure at once, even from up-close

Further author information: (Send correspondence to Sarthak Pathak) E-mail: pathak@robot.t.u-tokyo.ac.jp

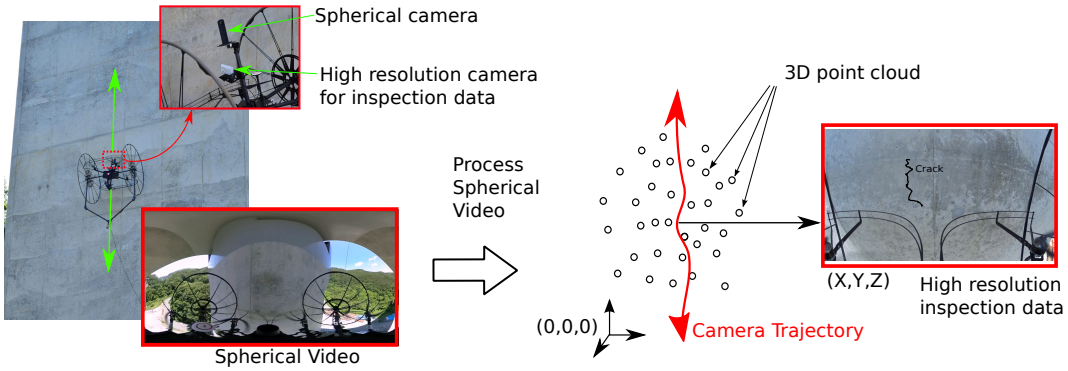


Figure 2. A UAV rolls on the surface of the bridge pillar in order to collect data from the surface and simultaneously records a spherical video. The spherical video is then processed to recover the 3D trajectory of the UAV is along with the sparse 3D structure, in order to map the inspection data to its correct location on the structure of the bridge pillar.

is particularly useful and advantageous for localization and reconstruction. An example of this is shown in Fig. 1 (b).

Thus, in this research, a spherical camera was used for UAV localization on a bridge pillar, via visual odometry. While collecting the inspection data, a spherical video was recorded and processed in order to estimate the camera trajectory and a sparse 3D point cloud of the pillar. The generated point cloud was then registered to the CAD model of the bridge[9] in order to map the inspection data.

Typical visual odometry methods adopt a strategy of detecting 2D feature points in an image frame, tracking them across images and reconstructing them in 3D. In planar images, all detected points have the same geometrical confidence. However, spherical images are not planar and exist as a sphere. Typical 2D image processing techniques which are necessary for visual odometry, such as feature-point detection and matching, only work on a 2D equirectangular projection, which contains a lot of distortion, making the vertically central portions of the image more reliable. If this is ignored during visual odometry calculations, large errors can occur as all parts of the image would contribute equally.

Thus, we take the image distortion and spherical image geometry into account during two essential visual odometry steps - the calculation of the 2-frame essential matrix via feature-point correspondences, and the registration of 3D points across 3 frames. In an experiment over a trajectory of 9.3 m on a real bridge pillar, our proposed improvements resulted in **localization error reduction from 0.69 m to 0.08 m**. The detailed method is described in the next section.

2. PROPOSED METHOD

2.1 Overview

In this research, a video taken from a spherical camera attached to a UAV is processed in order to estimate the camera motion and sparsely reconstruct the bridge pillar. In order to deal with the distortions, A-KAZE[10] is used for feature detection and matching. Our approach works on a moving window of frame triplets. Henceforth, we shall refer to the frames of this triplet as frame 1, 2, and 3.

Initially, features are tracked between frame 1 and frame 2. A filter is used to check whether frame 2 has sufficient translational displacement from frame 1, that is necessary for accurate estimation. The feature matches are filtered via RANSAC[11] and the essential matrix is estimated via a modified 8-point algorithm[12] which weighs the feature-point matches based on the distortion present near them. Following this, the pixels are then triangulated to 3D positions and frame 3 is processed. Feature matches that are common to all three frames are estimated and filtered on the basis of triangulated 3D points. The filtered points are registered based on a spherical reprojection error minimization to estimate the position of the frame 3. Upon finishing this estimation, the three-frame window is moved along the window sequence in order to process the entire video.

The intuition behind both steps is to take the spherical camera geometry and distortion into account during the visual odometry calculations. During the 2D feature matching, it is theorized that the distortion is inversely proportional to the "confidence" and thus directly proportional to the uncertainty of every feature point detected inside the equirectangular projection.

2.2 Two-view Estimation

In this section, we describe the two-view estimation carried out between consecutive frames. Typically, the two-view essential matrix of central-projective geometry satisfies the following equation.

$$\hat{\mathbf{x}}_2^T \; \mathtt{E} \; \hat{\mathbf{x}}_1 = 0, \tag{1}$$

where $\mathtt{E}$ is a matrix known as the *essential matrix* and $\hat{\mathbf{x}}_1$, $\hat{\mathbf{x}}_2$ are the unit vectors of the matched features in both images. It encodes the complete motion between the two images, i.e. the rotation and translation, up to a scale factor. It can be estimated linearly from 8 distinct feature matches. This is known as the *8-point* algorithm.[12] If corresponding spherical image points are written as $\hat{\mathbf{x}}_1 = [x_1, y_1, z_1]$ and $\hat{\mathbf{x}}_2 = [x_2, y_2, z_2]$, Eq. (1) can be re-written as:

$$\begin{bmatrix} x_2x_1, \; x_2y_1, \; x_2z_1, \; y_2x_1, \; y_2y_1, \; y_2z_1, \; z_2x_1, \; z_2y_1, \; z_2z_1 \end{bmatrix} \mathbf{e} = 0, \tag{2}$$

where $\mathbf{e} = [\mathtt{E}_{11}, \; \mathtt{E}_{12}, \; \mathtt{E}_{13}, \; \mathtt{E}_{21}, \; \mathtt{E}_{22}, \; \mathtt{E}_{23}, \; \mathtt{E}_{31}, \; \mathtt{E}_{32}, \; \mathtt{E}_{33}]^T$, a column vector containing all elements of $\mathtt{E}$.

Thus, for 8 different feature matches ${\hat{\mathbf{x}}_1}^n \rightarrow {\hat{\mathbf{x}}_2}^n$, $(n \in 1:8)$ the equation becomes:

$$\begin{bmatrix} {x_2}^1{x_1}^1, & {x_2}^1{y_1}^1, & {x_2}^1{z_1}^1, & {y_2}^1{x_1}^1, & {y_2}^1{y_1}^1, & {y_2}^1{z_1}^1, & {z_2}^1{x_1}^1, & {z_2}^1{y_1}^1, & {z_2}^1{z_1}^1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ {x_2}^8{x_1}^8, & {x_2}^8{y_1}^8, & {x_2}^8{z_1}^8, & {y_2}^8{x_1}^8, & {y_2}^8{y_1}^8, & {y_2}^8{z_1}^8, & {z_2}^8{x_1}^8, & {z_2}^8{y_1}^8, & {z_2}^8{z_1}^8 \end{bmatrix} \mathbf{e} = 0. \tag{3}$$

This set of linear equations condensed as $\mathtt{Ae} = 0$ can be solved by decomposing $\mathtt{A}$ using singular value decomposition to obtain the matrix $\mathtt{E}$. After recovering the correct $\mathtt{E}$ matrix, it can be decomposed to give the rotation matrix and translation vector.

In our modified approach, we multiply each row of Eq.(3) with the distortion rate w_{12}, as calculated below:

The distortion rate is calculated as follows. As seen in Fig. 3 (a), towards the equator of the image, the radius along the vertical axis is of unit length. Towards the top and bottom, the radius keeps decreasing to 0, but number of pixels in the corresponding v coordinate of the equirectangular image remains the same. Thus, the equirectangular image is stretched by a factor of $\frac{1}{r}$, where r is the radius of the sphere from the vertical axis. r can be calculated as:

$$\begin{aligned} r &= \sqrt{1 - z^2} \\ &= \sqrt{1 - \cos^2\left(\frac{\pi v}{h}\right)}. \end{aligned} \tag{4}$$

The following weight $w_s(v)$ is calculated to counter the distortion rate:

$$w_s(v) = \sqrt{1 - \cos^2\left(\frac{\pi v}{h}\right)}. \tag{5}$$

Thus, the distortion rate for each matched feature pair is calculated as:

$$w_{12} = w_s(v_1) \times w_s(v_2), \tag{6}$$

where $w_s(v_1)$ and $w_s(v_2)$ represent the individual distortion rates of the matched features in both images. The computation is done via RANSAC[11] filtering, based on an epipolar error as calculated on a spherical surface[13] . The essential matrix is decomposed to provide the two-view translation vector and rotation matrix, and the filtered feature matches are triangulated to 3D positions.

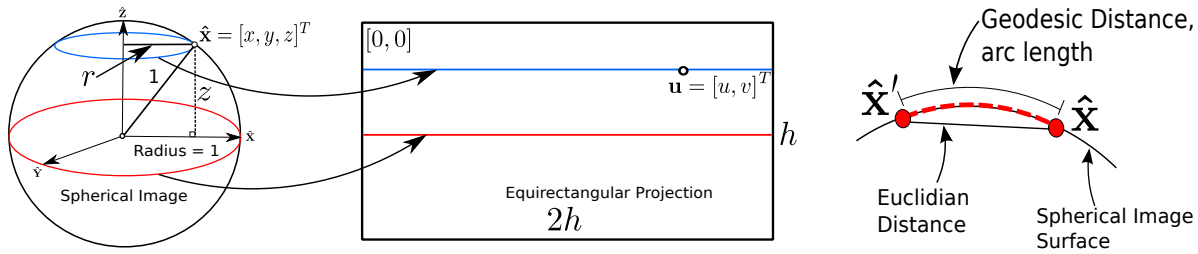


Figure 3. (a) Distortion of pixels inside a spherical image: The red and blue circles expand to the same length on the equirectangular projection. The distortion can be calculated according to the height of each pixel. (b) Spherical Epipolar Error: Euclidian distance is invalid for spherical image geometry as it does not lie on the spherical image surface.

2.3 Three-view Estimation

In the previous section, features were matched, RANSAC[11] filtered to calculate the two-view camera motion, and triangulated to 3D. In this section, frame 3 is localized by finding the correspondences between its features and the already triangulated 3D features from frame 1 and frame 2. This is done via a modified spherical reprojection error minimization.

Initially, the same two-view estimation described in the previous section is performed between frame 2 and frame 3. This provides two kinds of information - the filtered feature matches between frame 2 and frame 3, as well as an initial estimate for the localization of frame 3. The filtered feature matches are used to find the 2D-3D correspondences between the features of frame 3 and the triangulated 3D points from frame 1 and frame 2.

Next, the triangulated 3D points are projected to frame 3. Ordinarily, the reprojection error is calculated as the Euclidian distance between the original and the reprojected points. This is valid for regular planar images as the straight line joining the points lies on the image surface. However, in case of spherical images, this is invalid because the points are projected on a spherical surface, as seen in Fig. 3 (a). Hence, the reprojection error is calculated on the surface of the sphere as the length of the arc joining the two points i.e. the *geodesic distance* $e_g(\hat{\mathbf{x}}, \hat{\mathbf{x}}')$ between the reprojected 3D point $\hat{\mathbf{x}}'$ and its corresponding 2D feature $\hat{\mathbf{x}}$. This has been shown to lead to a better conditioned optimization and a more accurate convergence in.[13] It is calculated as follows:

$$e_g(\hat{\mathbf{x}}, \hat{\mathbf{x}}') = \sin^{-1}(\hat{\mathbf{x}}^T \hat{\mathbf{x}}'). \tag{7}$$

Thus, if $(t_{x,3}, t_{y,3}, t_{z,3}, \alpha_3, \beta_3, \gamma_3)$ represent the translation and rotation of frame 3 respectively w.r.t frame 2 in the 6 DoF Euler space space, the minimization of the reprojection is posed as follows:

$$\underset{t_{x,3}, t_{y,3}, t_{z,3}, \alpha_3, \beta_3, \gamma_3}{\text{minimize}} \sum_{\forall(\hat{\mathbf{x}})} \left(e_g(\hat{\mathbf{x}}, \hat{\mathbf{x}}')\right)^2. \tag{8}$$

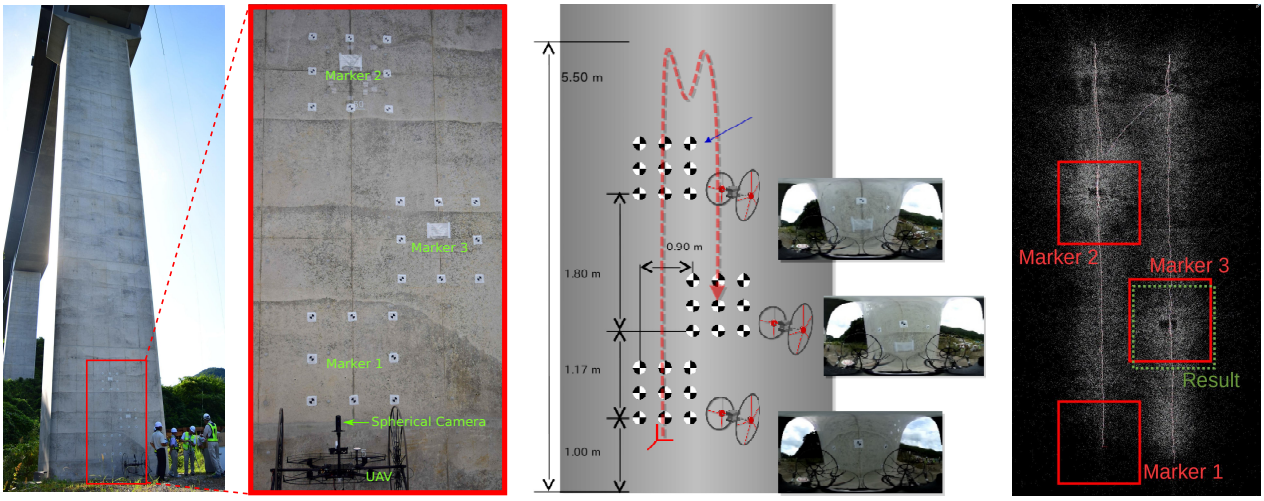
This minimization is solved using the Levenberg-Marquardt non-linear least squares minimization[14] in order to obtain the translation and rotation parameters for frame 3. This process is carried out over a moving 3-frame window over the entire video sequence. The obtained translation and rotation parameters are used to successively register the triangulated 3D points in order to obtain the structure of the bridge. The overall scale of the visual odometry process is set according to the translation between the first two frames. In the next section, we evaluate the effect of using the proposed improvements that consider the distortion of the equirectangular image.

3. EXPERIMENTAL EVALUATION

3.1 Experimental Setup

An experiment was conducted on a real bridge pillar in order to evaluate the localization accuracy. A UAV was fitted with a Ricoh Theta S spherical camera, as shown in Fig. 4(a). A bridge pillar with a height of 37.8 m was chosen for this experiment. Special, easily identifiable marker patterns were fitted at three known locations. The pillar used for evaluation and the locations of the markers are shown in Fig. 4 (a). The positions of the UAV while viewing each marker were estimated using the Omnidirectional Calibration Toolbox.[15] The position of the UAV at marker 1 was counted as the initial position estimate. The scale of motion was estimated using the length between the positions at marker 1 and marker 2. They were also used in order to calibrate the axes.

The UAV started from the ground crossing marker 1. It kept going upwards and crossed marker 2, till a height of around **5.5 m**. Then, the UAV drifted towards the right, and returned back to a position near marker 3. The trajectory is shown in Fig. 4(b).



(a) (b) (c)

Figure 4. (a) Experimental setup and environment (b) UAV Trajectory (c) Estimation result and ground truth positions of the three markers

3.2 Accuracy Evaluation

In order to evaluate the improvement in localization accuracy due to the proposed contributions, visual odometry was conducted with and without the proposed improvements and compared. In the case without the proposed improvements, the two-view estimation was done without the distortion rates and the reprojection error for the three-view estimation was done without spherical reprojection error minimization. The video was processed at a resolution of 1000 $\times$ 500 pixels. The trajectory consisted of a sequence of approximately 300 frames.

The difference between the expected position and the estimated position at marker 3 was estimated and taken to be the localization error in both approaches. The total length of this trajectory was around 11 m. The errors of estimation in both approaches are reported below in Tab. 1. The estimation result from the proposed approach is also shown in Fig. 4 (c). As can be seen from the results, the proposed improvements resulted in a reduction of 8.6 times in the error.

Table 1. Results of experiment 1

Method	**Error**	**Percentage Error**
Without Proposed Improvements	0.69 m	6.3 percent
With Proposed Improvements	0.08 m	0.7 percent

4. CONCLUSION

In this research, we proposed an improved visual odometry algorithm for spherical images, to provide UAV localization for a bridge inspection system. Two-view epipolar estimation was done by taking into account the distortion inside the equirectangular projection of the spherical image and three-view estimation was done by minimizing a spherical reprojection error. An experiment on a real bridge pillar resulted in localization error reduction by 8.6 times.

Acknowledgement

This work was in part supported by the Council for Science, Technology and Innovation, "Cross-ministerial Strategic Innovation Promotion Program (SIP), Infrastructure Maintenance, Renovation, and Management" (funding agency: NEDO).

REFERENCES

[1] Metni, N. and Hamel, T., "A uav for bridge inspection: Visual servoing control law with orientation limits," *Automation in construction* **17**, 3–10 (November 2007).

[2] Takahashi, N., Yamashita, S., Sato, Y., Kutsuna, Y., and Yamada, M., "All-round two-wheeled quadrotor helicopters with protect-frames for air-land-sea vehicle (controller design and automatic charging equipment)," *Advanced Robotics* **29**, 69–87 (January 2015).

[3] Hallermann, N. and Morgenthal, G., "Visual inspection strategies for large bridges using unmanned aerial vehicles (uav)," in [*Proceedings of 7th IABMAS International Conference on Bridge Maintenance, Safety and Management*], 661–667 (July 2014).

[4] Fathi, H., Dai, F., and Lourakis, M., "Automated as-built 3d reconstruction of civil infrastructure using computer vision: Achievements, opportunities, and challenges," *Advanced Engineering Informatics* **29**, 149 – 161 (April 2015).

[5] Brilakis, I., Fathi, H., and Rashidi, A., "Progressive 3d reconstruction of infrastructure with videogrammetry," *Automation in Construction* **20**, 884 – 895 (November 2011).

[6] Klein, G. and Murray, D., "Parallel tracking and mapping for small ar workspaces," in [*Proceedings of the IEEE and ACM International Symposium on Mixed and Augmented Reality*], 225–234 (November 2007).

[7] Forster, C., Pizzoli, M., and Scaramuzza, D., "Svo: Fast semi-direct monocular visual odometry," in [*Proceedings of the IEEE International Conference on Robotics and Automation*], 15–22 (May 2014).

[8] Engel, J., Sturm, J., and Cremers, D., "Semi-dense visual odometry for a monocular camera," in [*Proceedings of the IEEE International Conference on Computer Vision*], 1449–1456 (December 2013).

[9] Yoshimura, R., Date, H., Kanai, S., Honma, R., Oda, K., and Ikeda, T., "Automatic registration of mls point clouds and sfm meshes of urban area," *Geo-spatial Information Science* **19**, 171–181 (October 2016).

[10] Alcantarilla, P., Nuevo, J., and Bartoli, A., "Fast explicit diffusion for accelerated features in nonlinear scale spaces," in [*Proceedings of the British Machine Vision Conference*], 13.1–13.11 (September 2013).

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

[12] Hartley, R., "In defense of the eight-point algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19**, 580–593 (June 1997).

[13] Pagani, A. and Stricker, D., "Structure from motion using full spherical panoramic cameras," in [*Proceedings of the IEEE International Conference on Computer Vision (Workshops)*], 375–382 (November 2011).

[14] Lourakis, M., "levmar: Levenberg-marquardt nonlinear least squares algorithms in C/C++." [web page] `http://www.ics.forth.gr/~lourakis/levmar/` (Jul. 2004).

[15] Scaramuzza, D., Martinelli, A., and Siegwart, R., "A toolbox for easily calibrating omnidirectional cameras," in [*Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*], 5695–5701 (October 2006).