PLANETARY ROVER LOCALIZATION WITHIN ORBITAL MAPS

A. V. Nefian, X. Bouyssounouse, L. Edwards, T. Kim,
E. Hand, J. Rhizor, M. Deans, G. Bebis and T. Fong

Carnegie Mellon University, NASA Ames Research Center, University Nevada at Reno

ABSTRACT

This paper introduces an advanced rover localization system suitable for autonomous planetary exploration in the absence of Global Positioning System (GPS) infrastructure. Given an existing terrain map (image and elevation) obtained from satellite imagery and the images provided by the rover stereo camera system, the proposed method determines the best rover location through visual odometry, 3D terrain and horizon matching. The system is tested on data retrieved from a 3 km traverse of the Basalt Hills quarry in California where the GPS track is used as ground truth. Experimental results show the system presented here reduces by over 60% the localization error obtained by wheel odometry.

Index Terms— autonomous navigation, localization, visual odometry, horizon matching

1. INTRODUCTION

Planetary rover localization on past and current NASA missions relies on human expertise in matching rover camera views and orbital terrain maps (image and elevation). In addition, the location of rovers can occasionally be verified by spacecraft imagery [1], although this is not a common circumstance. Figure 1 illustrates several digital elevation models (DEM) reconstructed from the navcam stereo camera on-board the Mars Science Laboratory (MSL) rover that have been aligned over the Martian orbital DEM reconstructed from HiRISE mission satellite imagery data through stereo image processing [2]. The automated system described in this paper attempts to use the same information (orbital
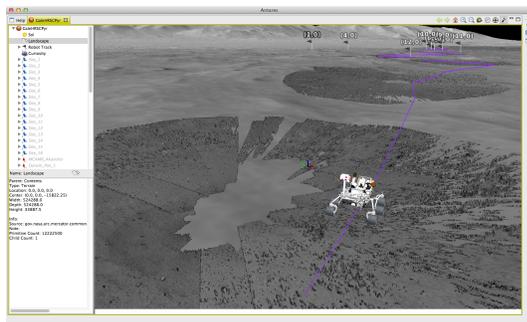

Fig. 1: MSL rover panorama localized over HiRISE terrain.

terrain maps and on-board rover imagery) to complement, enhance or (eventually) completely replace the current practice. A typical approach to image-based pose estimation is to register successive point clouds generated from stereo rover imagery by minimizing their square errors [3, 4] and integrating the relative poses sequentially. To reduce the error accumulation in consecutive pose estimation [5, 6, 7], most navigation systems take advantage of efficient bundle alignment [8, 9, 10, 11] and make use of Inertial Measurement Unit (IMU) and wheel odometry measurements [12]. Automated horizon matching algorithms have also been proposed to identify the location of a rover using a panoramic surface image and a DEM of the surrounding terrain [13, 14, 15].

In this paper we propose a novel unified approach for global pose estimation in the absence of GPS by minimizing a cost function that penalizes the errors between (1) the estimated rover pose and the pose predicted through visual odometry, (2) the 3D terrain from the estimated rover pose and the orbital terrain map, and (3) the horizon curve detected in the rover imagery and the horizon rendered from the estimated rover pose over the orbital terrain. The overall system is shown in Figure 2.The components of this block diagram are explained in more detail in the following sections.
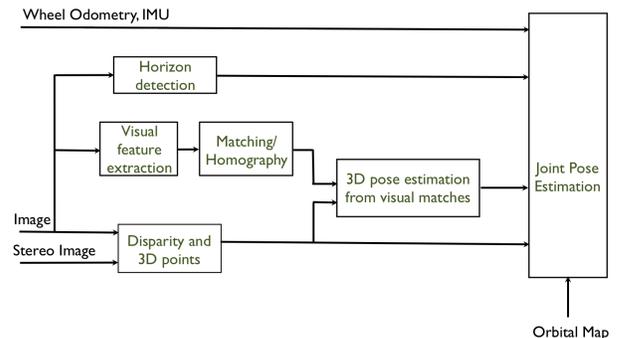

Fig. 2: The overall rover localization system.

2. TERRAIN RECONSTRUCTION

The stereo imagery obtained from the camera system [16] on board the rover is used to compute a set of 3D point clouds using the block matching algorithm implemented in

the OpenCV package [17]. The stereo camera system with a baseline of 30 cm and focal length of 3.5 mm is calibrated using the method described in [18]. Outliers in the reconstructed terrain are removed using a 3×3 morphological filter. Our current system achieves 5Hz on the full image resolution (1388×1038). It is used in visual stereo odometry (Section 3) and in supporting the localization framework (Section 5) through 3D terrain matching with the terrain model derived from stereo satellite imagery. The reconstructed point clouds are limited to a range of 30 m around the rover location. Points outside this range have a spatial resolution below the resolution of the orbital terrain model and often correspond to noisy data.

3. STEREO VISUAL ODOMETRY

Stereo visual odometry re-estimates the rover pose at every frame using the rover imagery [19] and the point clouds generated as described in Section 2. The stereo visual odometry system starts with the detection of SURF keypoints [20]. SURF keypoint extraction is faster than SIFT [21] keypoint extraction and allows for a larger and more reliable set of matches than obtained using ORB [22] keypoints. The keypoints extracted from the current image are matched against images from the previous frame using FLANN [23]. To reduce the number of false matches, the process is repeated by matching keypoints in the previous frame to keypoints in the current frame and selects only those keypoints and matches that pass both conditions. The remaining outlier matches are removed using the RANSAC algorithm [24] and by constraining the presence of a homographic transformation between matched keypoints in consecutive frames. The matched keypoints and their associated 3D position (Section 2) are used to estimate the current rover pose relative to its previous pose using the method described in [4]. The stereo visual odometry pose is estimated at 2Hz (same as the image capture rate) and is used in the localization framework described in Section 5. Figure 3 shows the a section of a 3D and texture mapped terrain model built using our stereo visual odometry approach.

4. HORIZON DETECTION AND RENDERING

Outside the range where terrain can be reconstructed from stereo imagery, visual features from the satellite imagery and rover imagery are often difficult to match due to variations in illumination conditions, camera viewing angle, unknown local albedo, and image resolution. However, the horizon curve in rover images remains a discriminant visual feature that enables global pose estimation by matching with the rendered horizon obtained from the orbital terrain map. The next subsections describe the horizon detection and rendering techniques used in our approach.
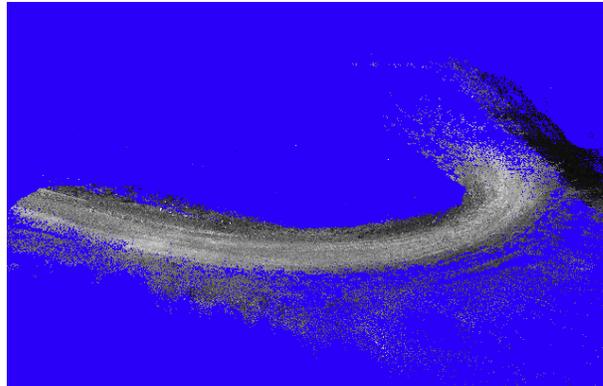

Fig. 3: Rover map obtained through visual odometry.

4.1. Horizon Detection

The statistical approach for horizon detection described in this paper does not use specific knowledge of the local terrain and sky intensity distribution and is meant to perform on a large variety of images collected from Earth, Mars, the moon, or other planetary bodies. Our method uses a Bayesian image formation model and a set of visual observations to determine the best image segmentation into ground and sky areas and thereby estimate the most likely horizon curve. Each pixel in the image is associated through the Bayesian model (Figure 4(a)) with an observed node (visual feature) and a hidden node with an unknown binary value that assigns it to sky or ground segments respectively.

The dense (per pixel) visual features used for horizon detection consist of the pixel grayscale intensity and local feature density. The local feature density is computed as the ratio of the number of local edges within a local window around each pixel and the area (in pixels) of the local window. In our particular implementation, the edges are computed using a 3×3 Sobel edge detector and the size of local window is chosen to be 7×7 pixels.

Let $P(O_{ij}|q_{ij})$ be the distribution of observation vector O_{ij} at pixel ij (column i, row j) in the image given the binary value (g:ground or s:sky) of the corresponding hidden node q_{ij} (Figure 4(a)). Figure 4(b) illustrates the initial distribution of visual features over ground (ground 1 and 2) and sky (clear sky and cloudy). High intensity areas (low and high feature density) are associated with cloud regions. High intensity features and low feature density areas are associated with sky. Those with low intensity features and low feature density are associated with terrain in heavy shadows (ground 2), and finally low image intensity features and high feature density areas are associated with regularly lit terrain. Let W and H be the width and height of the image and $P(\mathbf{O}_j|\mathbf{Q}_j)$ be the probability of the observation vectors $\mathbf{O}_j = O_{1j} \ldots O_{Hj}$ in image column j given the corresponding sequence of hidden nodes $\mathbf{Q}_j = q_{1j} \ldots q_{Hj}$. For observation vectors extracted by scanning each column from top to bottom we can assume without loss of generality that there is only one transition from sky to
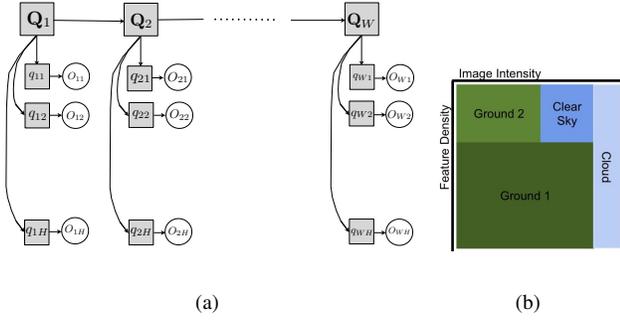
Fig. 4: (a) Bayesian image formation model. (b) $P(O_{ij}|q_{ij})$ sky (clear sky and cloudy sky) and terrain (ground 1 and 2) distribution over the space of visual features (image intensity and feature density).

ground and that a transition from ground to sky is constrained to not occur. Then, the observation probability in column j given that the transition from sky to ground occurs in pixel (k, j) is given by the equation

$$P(\mathbf{O}_j|\mathbf{Q}_j) = \prod_i^H P(O_{ij}|q_{ij}) \prod_{i=1}^k P(q_{ij} = s) \prod_{i=k+1}^H P(q_{ij} = g).$$

Furthermore, there are a total of H values associated with each $\mathbf{Q}_j$ sequence, one for each pixel in the column j where the unique transition from sky to terrain occurs. This specific pixel is the horizon pixel in column j. The set of all the horizon pixels across each column defines the horizon line.

An additional assumption of a smooth horizon line may be added, wherein jumps of more than N pixels between horizon pixels in consecutive columns are not allowed, with a transition probability between consecutive columns given by

$$P(\mathbf{Q}_j = k|\mathbf{Q}_{j-1} = l) = \begin{cases} \frac{1}{N} & \text{if } |k - l| \leq N \\ 0 & \text{otherwise} \end{cases}.$$

The conditional probability of the observed nodes given the hidden nodes then becomes

$$\begin{aligned} P(\underline{\mathbf{O}}|\underline{\mathbf{Q}}) &= P(\mathbf{O}_1 \dots \mathbf{O}_W|\mathbf{Q}_1 \dots \mathbf{Q}_W) \\ &= \prod_j^W P(\mathbf{O}_j|\mathbf{Q}_j) \prod_{j=1}^W P(\mathbf{Q}_j|\mathbf{Q}_{j-1}), \quad (1) \end{aligned}$$

where $\underline{\mathbf{O}} = \mathbf{O}_1 \dots \mathbf{O}_W$ and $\underline{\mathbf{Q}} = \mathbf{Q}_1 \dots \mathbf{Q}_W$. The probability of the pixel observations can be approximated by the above joint probability as shown by the following equation:

$$P(\underline{\mathbf{O}}) \approx \max_{\underline{\mathbf{Q}}} P(\underline{\mathbf{O}}, \underline{\mathbf{Q}}) = P(\underline{\mathbf{O}}|\underline{\mathbf{Q}})P(\underline{\mathbf{Q}}) \propto P(\underline{\mathbf{O}}|\underline{\mathbf{Q}}). \quad (2)$$

With the above formulation, finding the horizon curve is equivalent to finding the best sequence of $\mathbf{Q}_j$ and is computed as the maximum likelihood of Equation 2. This can be obtained efficiently via the Viterbi [25] algorithm. Figure 5 illustrates an example of the horizon curve detection (blue) in a rectified image captured by the on-board camera.

Fig. 5: Example of horizon curve (blue) detection over a rectified image taken by an on-board camera.

4.2. Horizon Rendering

Horizon rendering generates a synthetic view of the horizon as would be observed by the rover mounted camera. This view is based on the orbital DEM and the camera intrinsics and extrinsics. An accurate rendered horizon curve is often determined by remote topographical features captured in large coverage orbital terrain models. However, handling very large terrain surfaces at the resolution required for terrain matching can easily exceed computer memory limits. This is of particular concern when the processing is offloaded from the host computer to a Graphical Computational Unit (GPU) for fast rendering. To accommodate these constraints, a low coverage (.8×.8 km), high resolution (1 m per post) terrain model is augmented with one having large coverage (10×10 km) but low resolution (9 m per post). The resulting terrain model is split into a set of tiles with multiple subsampled resolution levels. This approach satisfies both the wide coverage requirements for horizon rendering and high resolution requirements for terrain matching, while accommodating the memory constraints of a typical GPU (around 1 GB for our processors). The rendered image is computed using OpenGL libraries [26], and the horizon curve is computed as the boundary of the rendered surface.

5. LOCALIZATION FRAMEWORK

Rover localization, or finding the optimal rotation ($\mathbf{R}$) and translation ($\mathbf{T}$) from a global reference point, is formulated as a cost function $\mathbf{L}$ minimization problem

$$\{\tilde{\mathbf{R}}, \tilde{\mathbf{T}}\} = \arg \min_{\{\mathbf{R}, \mathbf{T}\}} \mathbf{L}(\mathbf{R}, \mathbf{T}).$$

The localization cost function $\mathbf{L}(\mathbf{R}, \mathbf{T})$ is given by

$$\begin{aligned} \mathbf{L}(\mathbf{R}, \mathbf{T}) = & \ \omega_h(\mathbf{h}_d - h_r(\mathbf{R}, \mathbf{T}))^2 + \omega_e(\mathbf{e}_o - \mathbf{e}_r(\mathbf{R}, \mathbf{T}))^2 \\ & + \omega_w(\mathbf{x}_w - \mathbf{x}(\mathbf{R}, \mathbf{T}))^2 + \omega_v(\mathbf{x}_v - \mathbf{x}(\mathbf{R}, \mathbf{T}))^2, \end{aligned}$$

where $\mathbf{h}_d$ and $\mathbf{h}_r(\mathbf{R}, \mathbf{T})$ are the detected and rendered horizon curves (Section 4) respectively. $\mathbf{e}_o$ is the terrain elevation

of the orbital map and $\mathbf{e}_r(\mathbf{R}, \mathbf{T})$ is the terrain elevation determined from the point clouds obtained from the rover stereo camera (Section 2) and the rover estimated global position. $\mathbf{x}_w$ is the global position estimated from wheel odometry, $\mathbf{x}_v$ is the global position estimated from stereo visual odometry (Section 3) and $\mathbf{x}(\mathbf{R}, \mathbf{T})$ is the global position determined by the estimated global rotation and translation. The weights ω_h, ω_e, ω_w and ω_v are chosen such that $\omega_h + \omega_e + \omega_w + \omega_v = 1$ and represent the reliability of each of the horizon, terrain elevation, wheel odometry and visual odometry modules used in localization respectively. They are also chosen to normalize with various number of samples and vector sizes in each modality.

6. EXPERIMENTAL RESULTS

Data used to validate the results of this paper was gathered from field trials in Basalt Hills State Park, California during a 3 km traverse at an average speed of 0.8m/s. Images from the stereo camera system on board the rover described in Section 2 were captured at an average 2Hz. Figure 6 shows the reduction in localization error obtained using the proposed system versus the errors introduced by using a purely wheel odometry-based localization solution. The errors are calculated with reference to ground truth data obtained by GPS. Due to rover operation run-time requirements, the proposed localization system is executed every 300 frames as evidenced by to the large discontinuities in localization error seen in Figure 6. Across the traverse, the average localization error is reduced by over 60% compared to the wheel odometry solution.
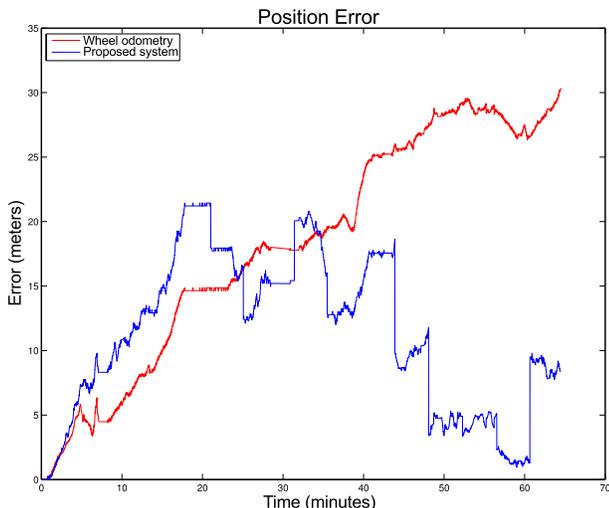

Fig. 6: Localization error for the proposed method (blue) and wheel odometry (red) vs. true GPS position.

Figure 7 shows the rover traverse ground truth determined by GPS (green), the estimated trajectories using wheel odometry (red), and the output of the proposed solution (blue).

Note that the differences between the GPS and wheel odometry trajectories increase from the starting point (top right in Figure 7) owing to accumulated error, while the proposed solution remains significantly closer to the GPS track during the entire traverse.
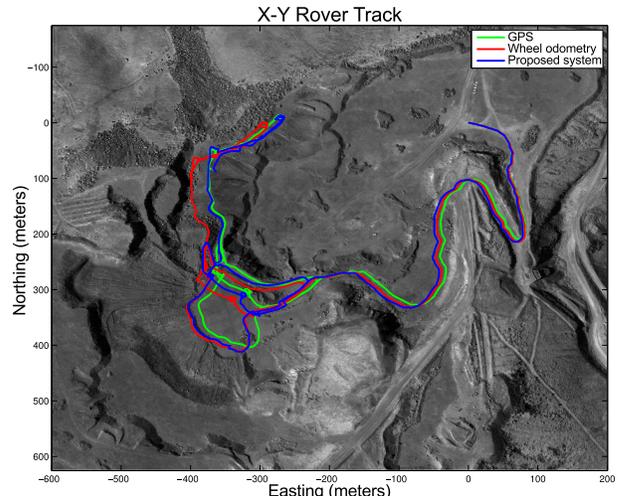

Fig. 7: Estimated rover track using the proposed method (blue), wheel odometry (red), and GPS (green).

7. CONCLUSIONS

The localization method presented in this paper combines visual odometry with terrain and horizon matching between rover and orbital views to accurately determine a rover's location within an orbital map. The accuracy of the method makes it suitable for autonomous or semi-autonomous robotic planetary exploration in the absence of pre-existing GPS or other beacon-based infrastructure. Future work will be directed towards a real time implementation of the localization system and testing with various rover configurations in a variety of terrestrial environments, as well as with imagery returned from current Mars missions.

8. REFERENCES

[1] Rongxing Li, Shaojun He, Yunhang Chen, Min Tang, Pingbo Tang, Kaichang Di, Larry Matthies, Raymond E Arvidson, Steven W Squyres, Larry S Crumpler, et al., "MER Spirit rover localization: Comparison of ground image–and orbital image–based methods and science applications," *Journal of Geophysical Research: Planets (1991–2012)*, vol. 116, no. E7, 2011.

[2] Mark Maimone, Yang Cheng, and Larry Matthies, "Two years of visual odometry on the mars exploration rovers," *Journal of Field Robotics*, vol. 24, no. 3, pp. 169–186, 2007.

[3] AJ Stoddart and A Hilton, "Registration of multiple point sets," in *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*. IEEE, 1996, vol. 2, pp. 40–44.

[4] Shinji Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 13, no. 4, pp. 376–380, 1991.

[5] Sebastien Gemme, David Gingras, Alessio Salerno, Erick Dupuis, François Pomerleau, and François Michaud, "Pose refinement using ICP applied to 3-D LIDAR data for exploration rovers," in *Proc. 2012 International Symposium on Artificial Intelligence, Robotics and Automation in Space*, 2012.

[6] Fabio Cozman, Eric Krotkov, and Carlos Guestrin, "Outdoor visual position estimation for planetary rovers," *Autonomous Robots*, vol. 9, no. 2, pp. 135–150, 2000.

[7] Luca Lucchese, Gianfranco Doretto, and Guido M. Cortelazzo, "A frequency domain technique for range data registration," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 11, pp. 1468–1484, 2002.

[8] Kurt Konolige and Willow Garage, "Sparse sparse bundle adjustment.," in *BMVC*, 2010, pp. 1–11.

[9] Niko Sünderhauf and Peter Protzel, "Towards using sparse bundle adjustment for robust stereo odometry in outdoor terrain," *Towards Autonomous Robotic Systems*, pp. 206–213, 2006.

[10] Michael Kaess, Ananth Ranganathan, and Frank Dellaert, "iSAM: Incremental smoothing and mapping," *Robotics, IEEE Transactions on*, vol. 24, no. 6, pp. 1365–1378, 2008.

[11] Michael Kaess, Hordur Johannsson, Richard Roberts, Viorela Ila, John J Leonard, and Frank Dellaert, "iSAM2: Incremental smoothing and mapping using the bayes tree," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 216–235, 2012.

[12] Rongxing Li, Kaichang Di, Larry H Matthies, Raymond E Arvidson, William M Folkner, and Brent A Archinal, "Rover localization and landing-site mapping technology for the 2003 mars exploration rover mission," *Photogrammetric engineering and remote sensing*, vol. 70, no. 1, pp. 77–90, 2004.

[13] Clark F Olson and Larry H Matthies, "Maximum likelihood rover localization by matching range maps," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*. IEEE, 1998, vol. 1, pp. 272–277.

[14] EE Palmer, RW Gaskell, LD Vance, MV Sykes, BK McComas, and WC Jouse, "Location identification using horizon matching," in *Lunar and Planetary Institute Science Conference Abstracts*, 2012, vol. 43, p. 2325.

[15] Georges Baatz, Olivier Saurer, Kevin Köser, and Marc Pollefeys, "Large scale visual geo-localization of images in mountainous terrain," in *Computer Vision–ECCV 2012*, pp. 517–530. Springer, 2012.

[16] Allied Vision Technologies, "Gige vision camera with Sony ICX267 and PoE option," .

[17] Gary Bradski, "The OpenCV library," *Doctor Dobbs Journal*, vol. 25, no. 11, pp. 120–126, 2000.

[18] Zhengyou Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*. IEEE, 1999, vol. 1, pp. 666–673.

[19] David Nistér, Oleg Naroditsky, and James Bergen, "Visual odometry," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. IEEE, 2004, vol. 1, pp. I–652.

[20] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, "SURF: Speeded up robust features," in *Computer Vision–ECCV 2006*, pp. 404–417. Springer, 2006.

[21] David G Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[22] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski, "ORB: an efficient alternative to SIFT or SURF," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2564–2571.

[23] Marius Muja and David G Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration.," in *VISAPP (1)*, 2009, pp. 331–340.

[24] Martin A Fischler and Robert 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.

[25] G David Forney Jr, "The Viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.

[26] Dave Shreiner and The Khronos OpenGL ARB Working Group, *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Versions 3.0 and 3.1*, Addison-Wesley Professional, 7th edition, 2009.