Opening the Black Box of 3D Reconstruction Error Analysis with VECTOR

Racquel Fygenson*†
Northeastern University

Kazi Jawad*‡
Wētā FX

Isabel Li
Art Center
College of Design

Francois Ayoub
Jet Propulsion Laboratory
California Institute of Technology

Robert G. Deen
Jet Propulsion Laboratory
California Institute of Technology

Scott Davidoff
Jet Propulsion Laboratory
California Institute of Technology

Dominik Moritz
Carnegie Mellon
University

Mauricio Hess-Flores
Jet Propulsion Laboratory
California Institute of Technology

ABSTRACT

Reconstruction of 3D scenes from 2D images is a technical challenge that impacts domains from Earth and planetary sciences and space exploration to augmented and virtual reality. Typically, reconstruction algorithms first identify common features across images and then minimize reconstruction errors after estimating the shape of the terrain. This *bundle adjustment* (BA) step optimizes around a single, simplifying scalar value that obfuscates many possible causes of reconstruction errors (e.g., initial estimate of the position and orientation of the camera, lighting conditions, ease of feature detection in the terrain). Reconstruction errors can lead to inaccurate scientific inferences or endanger a spacecraft exploring a remote environment. To address this challenge, we present VECTOR, a visual analysis tool that improves error inspection for stereo reconstruction BA. VECTOR provides analysts with previously unavailable visibility into feature locations, camera pose, and computed 3D points. VECTOR was developed in partnership with the Perseverance Mars Rover and Ingenuity Mars Helicopter terrain reconstruction team at the NASA Jet Propulsion Laboratory. We report on how this tool was used to debug and improve terrain reconstruction for the Mars 2020 mission.

Index Terms: Computer vision, stereo image processing, optimization, error analysis, uncertainty, SLAM, SfM, robotics.

1 INTRODUCTION

Stereo image processing to reconstruct a three-dimensional scene from two-dimensional images is an increasingly important capability in many domains. Such algorithms brings new analytic perspectives to fields as broad and diverse as geosciences [18], cultural heritage [13], archaeology [24, 10], Earth [11] and planetary science, and robotics for space exploration [12, 26, 25]. As a camera or set of cameras moves around a scene, perceived motion in the plane of the camera(s) (i.e., *parallax*) can be used to determine the distance of an object. As the same objects are viewed across multiple images, a powerful family of algorithms that includes simultaneous localization and mapping (SLAM) [28] and structure-from-motion (SfM) [3] can be used to build a 3D model of the scene.

These stereo reconstruction algorithms generally identify and match visual features, or **tiepoints**, corresponding to points in the scene (i.e., **tracks**) across all available images, and estimate a 3D point cloud via **triangulation**, given known or estimated camera positions and orientations (i.e., **pose**). These algorithms then run *bundle adjustment* [32] (**BA**) to minimize *total reprojection error* across many parameters– often thousands of cameras and 3D points [20, 22, 2]–and generate a single goodness-of-fit statistic. This single scalar value—which represents the complex set of interacting parameters in stereo reconstruction—provides no information about how individual residuals, which can be influenced

by inaccurate pose and structure estimates, impact the optimization process. This metric also provides no visibility into how particular images, lighting conditions, camera positions, or details of the morphology of the remote environment might interact to create inaccuracies in a particular output. The impact of these unknowns compounds in domains where high accuracy terrain reconstruction is critical to outcomes, like science or space exploration where there is no ground truth and inaccurate reconstruction can lead to false results or risking billion-dollar spacecraft. Alarmingly, while there has been an explosion of stereo algorithms across many fields, there has been little work visualizing errors of such processes and the effects of individual outliers on the accuracy of results.

In response, we developed VECTOR, a visual analytics application for the **V**isual **E**diting of **C**amera **T**iepoints, **O**rientations and **R**esiduals. VECTOR is an interactive visualization tool that shows the sensitivity of scene structure before BA (i.e., during feature tracking, pose estimation, and triangulation), as well as after BA (i.e., the interaction between all parameters), better characterizing model output, errors, and noise. Furthermore, VECTOR users can analyze outputs visually, which speeds up the debugging of algorithms. Thus, VECTOR accelerates algorithmic development and enables more informed operational decisions.

This paper presents VECTOR and its application to characterizing stereo reconstruction errors in mission operations for the Mars Perseverance Rover and Ingenuity Helicopter. In particular, this paper makes the following contributions:

1. Introduces the first visual analytics application to facilitate error analysis for stereo reconstruction algorithms.
2. Describes the process of error analysis for stereo reconstruction in the context of planetary robotic exploration.
3. Illustrates how visual encoding and interaction can address the challenges of terrain reconstruction.

2 BACKGROUND

In this section, we situate stereo reconstruction error analysis within the broader context of error analysis and uncertainty visualization in machine learning and robotics. We then unpack the challenges facing stereo reconstruction error analysis.

2.1 Understanding Machine Learning Output

VECTOR is motivated by a long history of using visualization and visual analytics to understand the performance of machine learning (ML) models. Recent work has focused on interactive explanation of model performance [16], or generation of rules [27] or counterfactuals [33] across broad classes of models from deep learning networks [17] to random forests [36], and in domains from bias detection [8] to space exploration [4] and the physical sciences [34].

Within machine learning, this paper focuses on the computer vision problem of stereo reconstruction, which has broad applicability across domains looking to recreate remote locations digitally [18, 13, 24, 10, 11, 12, 26, 25]. Research into error analysis of stereo reconstruction often focuses on mathematical ap-

---

*Kazi Jawad and Racquel Fygenson contributed equally to this work.

†e-mail: fygenson.r@northeastern.edu

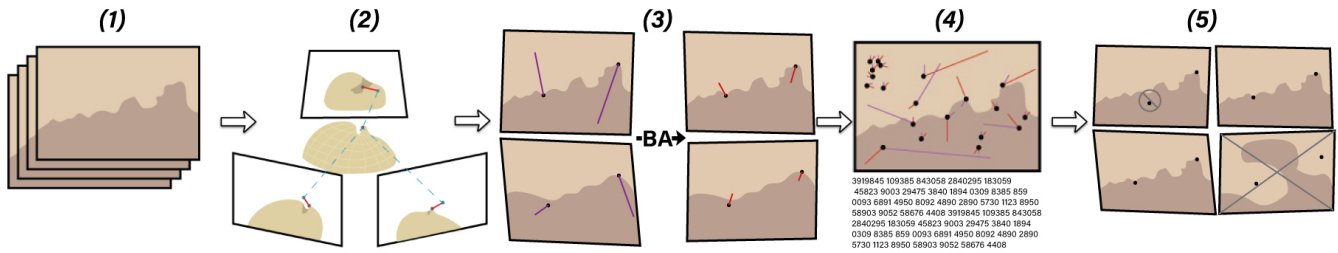‡e-mail: kjawad@wetafx.co.nz

Figure 1: The stereo scene reconstruction process. Starting with a set of 2D images for stereo reconstruction (step 1), the scene reconstruction algorithm computes tiepoints and tracks, triangulates their 3D position and back-projects them into the original images creating residuals (step 2). Then, the BA algorithm optimizes camera poses and 3D structure to minimize residuals (step 3), outputting optimized residuals in each image along with descriptive statistics in XML (step 4). Analysts then manually remove erroneous tracks (e.g., top-left of step 5) and camera poses (e.g., bottom-right of step 5), and re-run the process.

proaches [14, 7, 9, 35], including ground-truth-based camera pose estimations [30], geometric error extraction free of simplifying assumptions [19], or volumetric explanations for the total reprojection error in BA [29]. VECTOR builds on these mathematical approaches, serving as a visual aid to error analysis that allows users to unpack model outputs to pursue root causes of errors.

2.2 Error Analysis for Stereo Reconstruction Algorithms

Stereo systems for rover navigation typically receive as input a set of images (Fig. 1 step 1), then compute features and tracks across them (Fig. 1 step 2), and use linear algebra and optimization techniques to produce a map of the viewed scene as well as fine-tune the pose of the camera(s) mounted on the rover [25, 26] (Fig. 1 step 3) received via telemetry. Typically, reconstruction algorithms such as SfM and SLAM minimize total reprojection error of all computed points across all cameras via BA [32].

The accuracy of a multi-view reconstruction relies on accurate feature tracking and matching [23, 5]. Tiepoints can be computed using dense or sparse algorithms [23, 5, 6] and linked via tracks. Because errors accumulate in downstream computation, track matching accuracy strongly influences overall scene reconstruction fidelity [14]. Even robust estimation procedures [14] are sensitive to lighting conditions and occlusions. Due to these errors, computed 3D points do not reproject exactly onto their initial tiepoint positions in each image (see Fig. 1, step 2). Hence, the objective of BA is to adjust parameters in such a way that the total reprojection error of the 3D points with respect to their corresponding tiepoints in each camera is minimized. This error corresponds to the sum of squares of **residual errors** for each 3D point reprojected onto each image plane with respect to its corresponding tiepoint. This minimization can be achieved using algorithms such as *Levenberg-Marquardt* (LM) [21].

In the absence of ground-truth information, BA is the only valid geometrical evaluation of accuracy, despite its tendency to miring in local minima. Furthermore, BA outputs a single, minimized scalar, revealing little to nothing about the optimization process, which variables most affected its outcome, outliers that heavily skewed the process, and other valuable information which could be used to improve the accuracy of the reconstruction. For instance, if a small number of feature tracks with high residual error represent a large percentage of the total reprojection error, these tracks could be removed or corrected to allow BA to converge much faster, and with a greater probability of reaching the global minimum.

3 How Users Find Errors with VECTOR

To understand how stereo reconstruction scientists approach error analysis, over one year, we partnered with a team of experts at the NASA Jet Propulsion Laboratory. We conducted semi-structured

interviews, and created prototypes that we iteratively co-designed and evaluated [15]. Below, we detail the main user goals we uncovered, followed by how VECTOR's design facilitates each goal.

3.1 User Goals

The variance in residual magnitudes and directions across feature tracks and adjusted cameras' location and orientation can reveal important information about parameters that is more useful than total reprojection error and can be used to improve BA performance. Ideally, residuals should be of minimum length and evenly distributed in 2-dimensional orientation. Stereo scientists evaluate the performance of BAs by analyzing patterns in the obtained residuals, which can number in the thousands. Resolving residuals is supported by three main goals:

1. *Identify specific feature tracks, or portions of these tracks, that contribute to large BA errors.* Feature tracks identify objects or terrain across multiple images, and typical reconstructions contain thousands of tracks. High residuals in even a few feature tracks can skew precise modeling of entire objects or scenes, so stereo scientists often attempt to locate and remedy tracks with large residuals.

2. *Delete high-error feature tracks.* After identifying feature tracks with large residuals or a concentration of residuals pointing in one direction, scientists can opt to remove these tracks and re-run BA. Feature tracks are characterized via unique identification strings.

3. *Compare pre- and post- feature track removal models to evaluate improvement.* After finding and deleting error-prone feature tracks, scientists re-run BAs and investigate if the track deletion improved model performance. Prior to VECTOR, this was performed by toggling between long text files of each model's residuals–an intractable task for large datasets.

VECTOR has been key in understanding sources and magnitude of errors in SLAM, stereo, navigation and terrain-mapping methods at JPL, specifically for the Mars 2020 Perseverance rover and Ingenuity helicopter. VECTOR provides a cohesive platform for data filtering, editing, and reiterative analysis to address all three main user goals, adding more value to user workflows than equivalent non-interactive visualizations. In the following subsections, we describe each of VECTOR's panels and how they were used to discover and correct issues in Ingenuity *Flight 26*. This dataset consists of 122 images, where 11380 tracks were computed using a modified SIFT algorithm [23]. This dataset consists of straight linear motions of the helicopter as well as sharp turns, where triangulation is more likely to fail. In these areas, JPL scientists found that many ground points were computed incorrectly. Using VECTOR, operations teams removed outlier tracks and re-computed *Flight 26*'s 3D reconstruction in a matter of minutes, instead of sinking

VECTOR VIEW PANELS
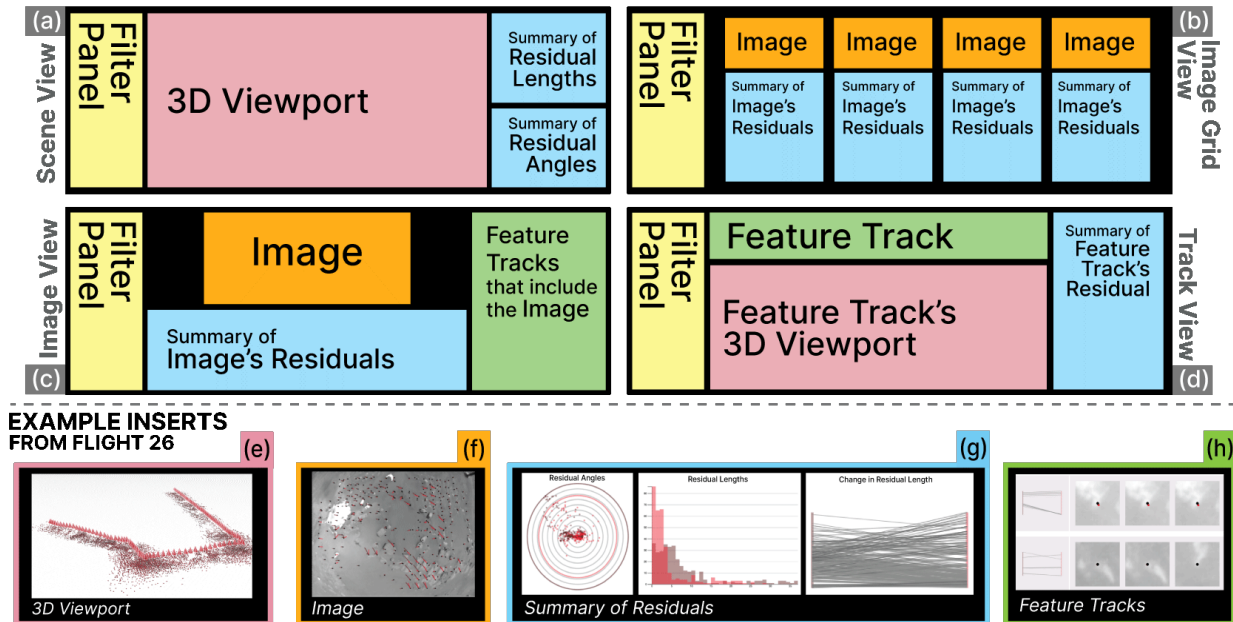

EXAMPLE INSERTS FROM FLIGHT 26

Figure 2: Top: VECTOR panels which are used in tandem to detect and eliminate erroneous feature tracks and camera poses that adversely affect BA in stereo reconstruction. Bottom: example visualizations populating the shown panels.

resources into parsing a large text file manually, as would have been done previously. This VECTOR use case was conducted on a Mac-Book Pro laptop using the Google Chrome web browser.

3.2 Data

The data for SLAM and stereo reconstruction at JPL typically consists of XML files. For a given dataset, one XML file contains camera pose parameters and another contains feature tracking pixel values, XYZ 3D positions, and other relevant metadata. These files can reach a few GB in size and are stored on clusters. Manually parsing such large files is slow and makes it virtually impossible to spot relationships between spatial variables and pinpoint error causes. With VECTOR, a user can import these XML files directly and see them visualized in 2D and 3D space, visually compare data from different BAs, and find and resolve problematic residuals.

3.3 VECTOR Panels

VECTOR consists of four views which highlight information about different parts of the reconstruction process. Each panel supports different pattern finding and visual investigation that are integral to the stereo operators' reiterative process. VECTOR's panels are designed with Shneiderman's Visual Information Seeking Mantra in mind: overview first, zoom and filter, then details-on-demand [31].

3.3.1 Scene View

VECTOR's entrypoint is the ***Scene View*** (Fig. 2a) which is a global representation of residuals across all tracks and cameras in the dataset. This view is broken down into an interactive BA viewport and a set of charts. The 3D viewport (Fig. 2e) represents bundle-adjusted camera frustums (conic shapes) and feature track points (dots), with a two-tone color palette that highlights whether elements are pre- or post- BA. Users can navigate the viewport using simple click-and-drag mouse movements as well as keyboard controls to pan, zoom, roll, and tilt. Users can also click on individual cameras or scene points to navigate to their respective ***Track Views*** or ***Image Views*** for further detail. If a scene point lies noticeably far from others (e.g., outlier points that are raised off the ground plane of Mars), users will likely analyze them directly in the Track

and Image Views for errors. Likewise, camera frustums that point in directions not consistent with their neighbors (e.g., Fig. 3a) could also indicate issues with the initial telemetry readings or with BA, and may be further analyzed individually. On the right-hand side of the Scene View, charts show summaries of residuals. As shown in Fig. 2g, a histogram encodes the distribution of residual lengths across the dataset and a radial chart depicts residuals' lengths and angles. The radial chart plots each residual vector with its origin in the center of the chart, encoding each vector's end point with a low-opacity point. These charts provide an overview of residual distributions which can be used to evaluate the general efficacy of a model, before investigating particular errors. Ideal BAs result in a concentration of short residuals (i.e., a left-skewed histogram) that are evenly distributed across all angles (i.e., circular, as opposed to oblong/oblate, clouds of points in the radial chart–see Fig. 3b).

For *Flight 26*, the visual information in the Scene View (Fig. 2e) displays sharp turns in the flight pattern. Upon panning and tilting the 3D scene, scientists noticed a set of 3D points which should lie on the surface of Mars were computed significantly off the plane (see Fig. 3c). Closer inspection reveals that these points coincide with regions where Ingenuity turned sharp corners. The short baselines in such cases have been shown to be ill-posed for 3D structure computation [14], and BA will generally fail to adjust these.

3.3.2 Image Grid View

The ***Image Grid View*** (Fig. 2b) is an alternative global representation to the Scene View. This view displays all images from the dataset in a horizontal carousel, along with each image's summary charts. Each image is shown at the top of its vertical card, and is superimposed with pre- and post-BA residuals. Below the image, a histogram and radial chart (e.g., Fig. 2g), similar to those in the Scene View, represent the distribution of residuals in each image. At the bottom, a slope chart highlights the change in individual residual lengths from pre- to post- BA (e.g., Fig. 2g, far right). This view allows users to quickly visually parse through each image and identify those that might be associated with large residuals indicative of low-quality images or tracks that can be removed for more accurate stereo modeling. From this view, users can click on images

camera after BA

(a)

(b)

Oblate radial chart: residuals tend to point in two directions

Points are uniformly distributed around the center of the chart.

camera before BA

(c)

Upward slopes indicate residuals increased with new BA

Downward slopes indicate residuals decreased with new BA

old BA   *new BA*   *old BA*   *new BA*
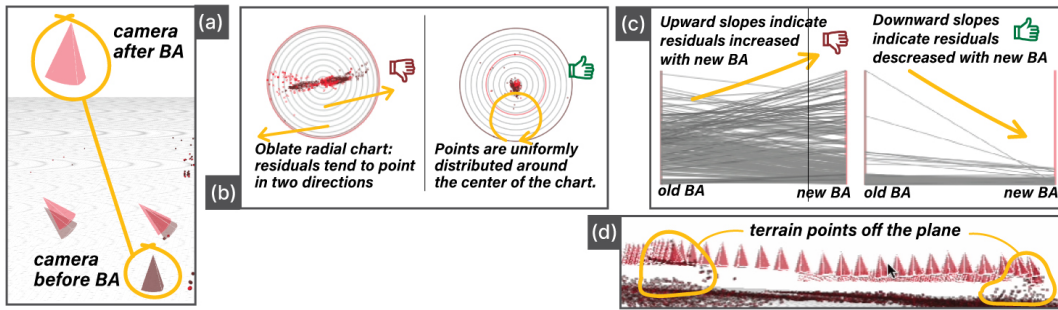
(d)

terrain points off the plane

Figure 3: Annotated example visualizations from *Flight 26*. (a) An example camera (circled) does not align with others. It could be removed to improve BA accuracy. (b) Ideally, residuals are short and uniformly distributed in angle. (c) Ideally, residuals decrease when a new BA is applied. (d) Inaccurate computed terrain points which should lie on the plane.

of interest to visit their individual **Image Views**.

For *Flight 26*, inspection of images revealed some higher residual errors **after** BA, which is typically due to bad triangulations of specific feature tracks that skew the optimization. Post-BA values on the right-hand axis of the residual slope diagram were higher than their pre-BA counterparts on the left (see Fig. 3d, left). Next, we entered the *Image View* panel to further investigate the tracks.

3.3.3   Image View

The **Image View** (Fig. 2c) represents the residual information for an individual image and all the feature tracks to which its residuals contribute. This view can be accessed by selecting a BA camera (cone-like prism in Fig. 2e) in the Scene View or an image card in the Image Grid View. In the Image View, each imported photo is superimposed with its corresponding residuals (e.g., Fig. 2f). For closer inspection of the image and residuals, users can zoom using trackpad pinch-to-zoom motions or adjusting a computer mouse scroll wheel. Users can also pan via simple click-and-grab motions. Like in the Image Grid View, a histogram and radial chart depicts an overview of the distribution of error vectors in the image. On the right-hand side of this view, every feature track that is built using the individual image in this view is shown in a scroll-able, horizontal card. On the far left of each feature track card, is a slope chart that encodes the change in error vector lengths between pre- and post-BA for the error vectors relevant to that track. To provide further 2D context, to the right of each slope chart sit a set of cropped images centered around the residual points that coalesce to create the track's single BA feature (e.g., Fig. 2h). The user can click on a horizontal track card to pull up its corresponding Track View. For *Flight 26*, clicking on a specific image indeed revealed that certain feature tracks had high errors. In Fig. 2c, the directional pattern of the bright red residual lines shows that the camera's helicopter was turning through a tight corner when it took this image. The consistency of the lines' angle and length confirmed that BA failed to minimize residuals. To check that these residuals resulted from a tracking error, not an issue with the camera models, using the *Track View*'s 3D view port (e.g., Fig. 2e), we saw that the cameras were correctly pointed towards the ground. Finally, we were able to delete these erroneous tracks directly in the Image View panel, and re-run the stereo reconstruction process. This entire process took minutes and yielded much-improved results.

3.3.4   Track View

Lastly, the **Track View** (Fig. 2d) depicts all residual information for an individual track, representing one triangulated point. The Track View can be accessed by clicking on a BA point in the Scene View or a horizontal track card on the right side of the Image View. The Track View provides granular information about each residual that contributes to a singular 3D feature. At the top of this view, each relevant residual is superimposed on top of its respective 2D image

for spatial context. Below these images, a BA viewport visualizes the camera poses that contribute to each residual, along with the single point that has been triangulated. Lastly, a set of summary charts similar to those in the previous views display residuals' distribution for quick overview of track accuracy. In the Track View, users can also opt to delete points and their feature tracks if they are characterized by egregious residuals.

3.3.5   Filtering Panel

The **Filtering Panel** lives on the left-hand side of all of VECTOR's views. If a user alters the parameters in one view, those parameters persist to other views. Filters only reset upon loading a new dataset or relaunching the system. Every view has a set of standard residual filters associated with type, length, angle, precision and scale. The type filter allows users to view specific residual groups, determined by the dataset. The length, angle, precision, and scale filters directly manipulate residual values shown in the view. The Scene View has additional filters for toggling rendered points or camera frustums. The Image and Image Grid View include additional sorting parameters for controlling the layout of tracks or images, respectively.

4   CONCLUSIONS AND NEXT STEPS

This paper presents the VECTOR (Visualization and Editing of Camera Tiepoints, Orientations, and Residuals) software which is currently used at NASA-JPL for evaluation and error analysis of stereo algorithms in its ground-based data processing pipeline. VECTOR was born out of a need to visualize errors and reconstruction parameters in common 3D stereo processes such as SfM and SLAM, and is being used for operations in the Mars 2020 mission.

Future work could improve VECTOR with the development of validation metrics to increase certainty about optimization. While current VECTOR users report major improvements in their ease of error finding, there is no way for users to know with certainty if errors have reached a global minimum. VECTOR was also tested on JPL datasets with hundreds of images and thousands of tracks, but scaling to datasets in the millions will need to be explored. Future work could also identify other domains that may benefit from VECTOR, within the vast amount of stereo applications [18, 13, 24, 10, 11, 12, 26, 25] and other difficult optimization problems. To this end, the software is open-source for public use at NASA-AMMOS.

4

REFERENCES

[1] Data to discovery visualization program. Online; accessed 29-April-2024. 4

[2] S. Agarwal, K. Mierle, and T. C. S. Team. Ceres Solver, 10 2023. 1

[3] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski. Building rome in a day. In *2009 IEEE 12th International Conference on Computer Vision*, pp. 72–79. IEEE, Sept. 2009. doi: 10.1109/ICCV.2009.5459148 1

[4] S. Bae, F. Rossi, J. V. Hook, S. Davidoff, and K.-L. Ma. A visual analytics approach to debugging cooperative, autonomous multi-robot systems' worldviews. In *2020 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 24–35. IEEE, New York, NY, USA, Oct. 2020. doi: 10.1109/VAST50239.2020.00008 1

[5] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In A. Leonardis, H. Bischof, and A. Pinz, eds., *Computer Vision – ECCV 2006*, vol. 3951 of *Lecture Notes in Computer Science*, pp. 404–417. Springer Berlin / Heidelberg, 2006. doi: 10.1007/11744023_32 2

[6] S. S. Beauchemin and J. L. Barron. The computation of optical flow. *ACM Computing Surveys*, 27(3):433–466, Sept. 1995. doi: 10.1145/212094.212141 2

[7] M. J. Brooks, W. Chojnacki, D. Gawley, and A. van den Hengel. What value covariance information in estimating vision parameters? In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, ICCV-01, pp. 302–308. IEEE Comput. Soc, 2001. doi: 10.1109/ICCV.2001.937533 2

[8] A. A. Cabrera, W. Epperson, F. Hohman, M. Kahng, J. Morgenstern, and D. H. Chau. Fairvis: Visual analytics for discovering intersectional bias in machine learning. In *2019 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 46–56. IEEE, Oct. 2019. doi: 10.1109/VAST47406.2019.8986948 1

[9] L. Cheong, C. Fermüller, and Y. Aloimonos. Effects of errors in the viewing geometry on shape estimation. *Computer Vision and Image Understanding*, 71(3):356–372, Sept. 1998. doi: 10.1006/cviu.1997.0649 2

[10] M. Dellepiane, N. Dell'Unto, M. Callieri, S. Lindgren, and R. Scopigno. Archeological excavation monitoring using dense stereo matching techniques. *Journal of Cultural Heritage*, 14(3):201–210, May 2013. doi: 10.1016/j.culher.2012.01.011 1, 4

[11] B. A. DeWitt and P. R. Wolf. *Elements of Photogrammetry (with Applications in GIS)*. McGraw-Hill Higher Education, 2000. doi: 10.1017/cbo9780511535062.018 1, 4

[12] G. Grisetti, R. Kümmerle, C. Stachniss, and W. Burgard. A tutorial on graph-based slam. *IEEE Intelligent Transportation Systems Magazine*, 2(4):31–43, 2010. doi: 10.1109/MITS.2015.2475018 1, 4

[13] G. Guidi, J.-A. Beraldin, and C. Atzeni. High-accuracy 3-d modeling of cultural heritage: The digitizing of donatello's "maddalena". *IEEE Transactions on Image Processing*, 13(3):370–380, Mar. 2004. doi: 10.1109/TIP.2003.822592 1, 4

[14] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2nd ed., Mar. 2004. doi: 10.1017/CBO9780511811685 2, 3

[15] M. Hendrie, H. Mushkin, S. Lombeyda, and S. Davidoff. JPL / Caltech / ArtCenter: Towards a collaborative methodology for interactive scientific data visualization. *Information Design Journal*, 27(1):76–84, 2022. doi: 10.1075/idj.22009.hen 2, 4

[16] F. Hohman, A. Head, R. Caruana, R. DeLine, and S. M. Drucker. Gamut: A design probe to understand how data scientists understand machine learning models. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, p. 1–13. ACM, New York, NY, USA, May 2019. doi: 10.1145/3290605.3300809 1

[17] F. Hohman, M. Kahng, R. Pienta, and D. H. Chau. Visual analytics in deep learning: An interrogative survey for the next frontiers. *IEEE Transactions on Visualization and Computer Graphics*, 25(8):2674–2693, Aug. 2018. doi: 10.1109/TVCG.2018.2843369 1

[18] M. R. James and S. Robson. Straightforward reconstruction of 3d surfaces and topography with a camera: Accuracy and geoscience application. *Journal of Geophysical Research: Earth Surface*, 117(F3), Aug. 2012. doi: 10.1029/2011JF002289 1, 4

[19] D. Knoblauch, M. Hess-Flores, M. Duchaineau, and F. Kuester. *Factorization of Correspondence and Camera Error for Unconstrained Dense Correspondence Applications*, pp. 720–729. Springer Berlin Heidelberg, 2009. doi: 10.1007/978-3-642-10331-5_67 2

[20] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics*, 2(2):164–168, 1944. doi: 10.1090/qam/10666 1

[21] M. Lourakis and A. Argyros. The design and implementation of a generic sparse bundle adjustment software package based on the Levenberg-Marquardt algorithm. Technical Report 340, Institute of Computer Science - FORTH, Heraklion, Crete, Greece, August 2000. 2

[22] M. A. Lourakis and A. Argyros. Sba: A software package for generic sparse bundle adjustment. *ACM Transactions on Mathematical Software*, 36(1):1–30, Mar. 2009. doi: 10.1145/1486525.1486527 1

[23] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov. 2004. doi: 10.1023/b:visi.0000029664.99615.94 2

[24] M. Magnani, M. Douglass, W. Schroder, J. Reeves, and D. R. Braun. The digital revolution to come: Photogrammetry in archaeological practice. *American Antiquity*, 85(4):737–760, Aug. 2020. doi: 10.1017/aaq.2020.59 1, 4

[25] M. Maimone, Y. Cheng, and L. Matthies. Two years of visual odometry on the mars exploration rovers. *Journal of Field Robotics*, 24(3):169–186, 2007. doi: 10.1002/rob.20184 1, 2, 4

[26] L. Matthies, M. Maimone, A. Johnson, Y. Cheng, R. Willson, C. Villalpando, S. Goldberg, A. Huertas, A. Stein, and A. Angelova. Computer vision on mars. *International Journal of Computer Vision*, 75(1):67 – 92, Mar. 2007. doi: 10.1007/s11263-007-0046-z 1, 2, 4

[27] Y. Ming, H. Qu, and E. Bertini. Rulematrix: Visualizing and understanding classifiers with rules. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):342–352, 2019. doi: 10.1109/TVCG.2018.2864812 1

[28] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós. Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015. doi: 10.1109/TRO.2015.2463671 1

[29] S. Recker, M. Hess-Flores, M. A. Duchaineau, and K. I. Joy. Visualization of scene structure uncertainty in a multi-view reconstruction pipeline. In *2012 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pp. 183–190. IEEE, Oct. 2012. doi: 10.1109/AIPR.2012.6528216 2

[30] V. Rodehorst, M. Heinrichs, and O. Hellwich. Evaluation of relative pose estimation methods for multi-camera setups. In *International Archives of Photogrammetry and Remote Sensing (ISPRS '08)*, pp. 135–140. Beijing, China, 2008. 2

[31] B. Shneiderman. The eyes have it: a task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE Symposium on Visual Languages*, pp. 336–343, 1996. doi: 10.1109/VL.1996.545307 3

[32] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. *Bundle Adjustment — A Modern Synthesis*, pp. 298–372. Springer Berlin Heidelberg, London, UK, 2000. doi: 10.1007/3-540-44480-7_21 1, 2

[33] J. Wexler, M. Pushkarna, T. Bolukbasi, M. Wattenberg, F. B. Viégas, and J. Wilson. The what-if tool: Interactive probing of machine learning models. *IEEE Transactions on Visualization and Computer Graphics*, 26:56–65, 2019. doi: 10.48550/arXiv.1907.04135 1

[34] A. P. Wright, P. Nemere, A. Galvin, D. H. Chau, and S. Davidoff. Lessons from the development of an anomaly detection interface on the mars perseverance rover using the ishmap framework. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, IUI '23, p. 91–105. ACM, New York, NY, USA, 2023. doi: 10.1145/3581641.3584036 1

[35] W. Zhao and N. Nandhakumar. Effects of Camera Alignment Errors on Stereoscopic Depth Estimates. *Pattern Recognition*, 29(12):2115–2126, December 1996. doi: 10.1016/S0031-3203(96)00051-9 2

[36] X. Zhao, Y. Wu, D. L. Lee, and W. Cui. iforest: Interpreting random forests via visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):407–416, 2019. doi: 10.1109/TVCG.2018.2864475 1