Planning Views to Model Planetary Pits under Transient Illumination

Heather Jones
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
heatherljones@cmu.edu

Wennie Tabib
Department of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
wtabib@cmu.edu

William L. "Red" Whittaker
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
red@cmu.edu

Abstract—**This paper addresses the problem of planning views for modeling large, local, substantially 3D terrain features at long range from surface rovers. These include building-size and stadium size pits with vertical walls. Pits have been identified in recent high-resolution images of the Moon and Mars. Planetary pits are interesting scientific targets created by collapse, often exposing layers of bare rock in their walls, hinting at past volcanism and other subsurface processes with their morphology. Some offer glimpses into caves. This paper presents a pipeline for view trajectory planning that enables detailed modeling of planetary pits from surface rovers. Techniques for converting prior terrain knowledge into a planning problem are developed, methods for planning rover images are discussed, and a comparison of different image-based reconstruction methods for pit modeling is presented. Results from preliminary field experiments for the end-to-end view trajectory planning pipeline are presented.**

TABLE OF CONTENTS

1. INTRODUCTION

This paper addresses the problem of planning views for modeling large, local, substantially 3D terrain features at long range from surface rovers. These include building-size and stadium size pits with vertical walls. Pits have been identified in recent high-resolution images of the Moon and Mars. Planetary pits are interesting scientific targets created by collapse, often exposing layers of bare rock in their walls, hinting at past volcanism and other subsurface processes with their morphology. Some offer glimpses into caves. Pits have flat or sloping aprons that drop from the surface to meet steep walls. These differ from the raised rims and blocky ejecta around craters. While surface rovers cannot negotiate steep pit walls, aprons facilitate long-range cross-pit viewing of far walls.

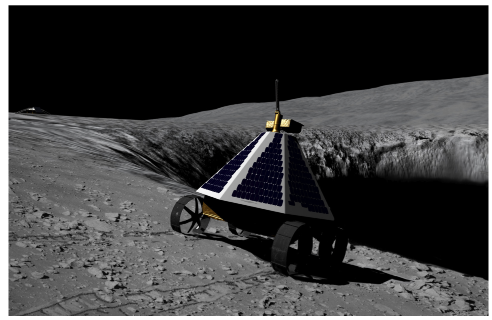


Figure 1. A rover views across a planetary pit to image the far wall.

Past planetary rovers have been myopic, sensing as needed to plan far enough to safely take the next step and examining nearby targets as they encounter them. In this case, a rover can be considered co-located with the viewed terrain. The pit modeling case, where targets lie on the un-reachable far wall of a pit, breaks this assumption. Multiple targets will be visible from a single position, and multiple positions will have views of the same target.

The quality of rover images is also greatly affected by the incidence angle of transiting sunlight, so the timing of images matters. Myopic rover operations, only planning within a rover's sensor view, would take many days to model a pit on Mars. On the Moon, where multiple daylight periods are not guaranteed, such myopic operations might fail to complete the task. The need is for a planner that can autonomously determine what targets to view, from what positions, at what times.

Rovers in this research are assumed to have at least one high-resolution camera. With the right lensing, cameras can capture high-resolution data at long distances. They can be used to image pit walls, reconstruct pit geometry, and even determine material types, but in order to get high quality models from camera images, attention must be paid to both the time-dependent illumination angles and the position-dependent view angles. This includes relative angles among sets of images used for stereo reconstruction or material identification. The ordering of views is also important, not only because of the time of day that images are taken, but also to minimize distance traveled.

The paper formulates the problem as a new vehicle routing problem: the Orienteering Problem with Time Windows and Inter-Node Dependencies (OPTWIND). This is related to the Orienteering Problem, an established vehicle routing problem, in which a traveler gets reward by visiting nodes and wants to maximize reward but at the same time minimize distance traveled. In the extension with Time Windows, visits

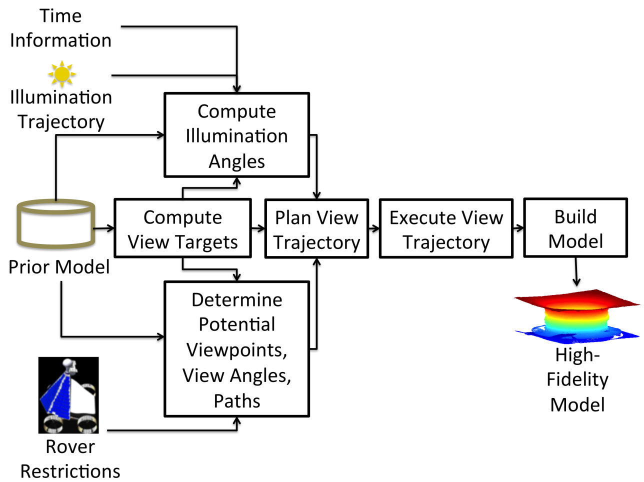


Figure 2. The pipeline for view trajectory planning.

to a node are limited to a certain time window. OPTWIND extends the Orienteering Problem with Time Windows, adding Inter-Node Dependencies.

This paper describes a pipeline for view trajectory planning for modeling a planetary pit from a surface rover, going from inputs available prior to a mission through the processing of mission data into a detailed pit model. King's Bowl pit in the Craters of the Moon National Monument and Preserve, Idaho, is used as an illustrative example.

The View Trajectory Planning Problem

The view trajectory planning problem selects what to image, from where, and at what time in order to build a good model. In order to plan, real-world inputs must first be translated into a format that the planner can understand, and after the view trajectory is executed and images are captured, these images must be combined to build a model. Fig. 2 shows a diagram of the pipeline for view trajectory planning for modeling pits. Inputs include a prior model of pit geometry, (however crude), information about the time (start and end) during which the modeling task must be completed, the trajectory of illumination direction over time for the pit to be modeled, and rover operating restrictions.

Example Cases

Because pits are the only cave entrances identified so far on the Moon and Mars, pits may one day provide entrances to caves. Reconnaissance modeling with a surface rover can help design and plan missions to descend into planetary caves.

The Lacus Mortis pit, located at 44.962 N and 25.61 E in the Moon's Lacus Mortis region, is approximately 110 m wide and 90 m deep and features a long ramp. This pit is used as an example for prior model construction in Section 4.

Indian Tunnel is a lava tube cave in Idaho's Craters of the Moon National Monument and Preserve. It has several skylight pits, one of which is modeled from imagery in Section 8. Evidence of lava tubes has been identified on the surface of both the Moon and Mars [1], [2].

King's Bowl pit is located along Idaho's Great Rift, in the Craters of the Moon National Monument and Preserve. The pit was formed due to a steam explosion along the rift. It is approximately 76 m long, 30 m wide and 30 m deep. At one end, the pit provides entrance into a deeper cave. While the formation mechanisms of pits on the Moon are likely different from this pit, pits associated with rift systems have been identified on Mars [2]. This pit also provides a good functional analog for planetary pit modeling. Rock layers can be identified in the pit walls, and the identification of shape and material properties for pit wall layers is a useful science objective, facilitating study of the formation of pits and their surrounding geological environments. King's Bowl is used as an example for steps throughout the view trajectory planning pipeline.

Paper Organization

The paper is organized as follows. Section 2 discusses related work in the areas of view planning, mobile robot planning, vehicle routing, and modeling. Section 3 presents the formulation of the OPTWIND problem. Section 4 through Section 7 discuss elements of the view trajectory planning pipeline. Section 4 discusses what is expected as input to the view trajectory planning problem. Section 5 discusses how these inputs are transformed into an instance of the view trajectory planning problem. Section 6 addresses the planning process. Section 7 compares several methods for image-based reconstruction, operating on the same set of King's Bowl images. Section 8 discusses results from modeling King's Bowl, with planned sets of images, and results from modeling a lava tube skylight pit. Section 9 summarizes the results. Section 10 concludes the document and presents ideas for future work.

2. Related Work

View planning is fundamental to view trajectory planning for modeling large, local terrain features like planetary pits with images taken by surface rovers. This is especially true in the discussion of how views are represented and how potential view-viewpoint pairs are selected. Unlike much view-planning work motivated by factory applications, a rover is constrained to move on the ground and avoid obstacles. Traveling between viewpoints takes time and incurs risk. This motivates the distinction between many view planning methods and view planning for mobile robots, including view trajectory planning. Planning routes with multiple destinations is also critical to view trajectory planning for pit modeling. The Orienteering Problem, and its multiple extensions, provide insight on how to formalize view trajectory planning for pit modeling. This section discusses prior methods that address view planning, view planning from mobile robots, and the Orienteering Problem and its extensions.

An understanding of how the model will be built from the planned images is also critical when planning images for model building. Current methods used to build models from planetary mission data are discussed, along with techniques for building models from many images, and techniques for taking advantage of illumination.

View Planning

Planning a view trajectory requires a representation of view. This is informed by prior work in view planning. What a view sees, *view target*, where it is seen from, *viewpoint*, (camera position and orientation), and how a view is created are all important.

Tarabanis, Allen and Tsai make the distinction between generate and test view planning methods, where a set of viewpoint parameters are generated and tested to determine the view target, and synthesis methods, where viewpoint parameters are calculated for each given view target [3]. They identify three components of the view planning problem: the sensor parameters, the sensor and object models, and the feature detectability constraints. Under sensor parameters, they include sensor and illumination position and orientation, as well as more camera-specific parameters (e.g. aperture, focal length, exposure time). In this research, only sensor position, sensor orientation, and illumination angle are considered. Camera-specific parameters are considered to be fixed or are automatically computed on the fly. This reduces the dimensionality of the planning problem.

Sensor and object models may contain models of image formation, geometry of the target object, and photometric information about the target material [3]. This work uses a pinhole camera model for planning and assumes that there is a coarse prior model of the terrain feature. Constraints on illumination and view angles, as used in this work, could implicitly encode knowledge of a terrain material. For example, if for the set of materials from which the feature is expected to be composed, there are known illumination angle ranges that will not produce good images, those ranges can be excluded by absolute illumination angle constraints.

Feature detectability constraints include whether there is line of sight to the feature, and whether the sensor field of view is sufficient to have the feature in frame, focus, resolution, and distortion [3]. In planning for pit modeling, line of sight, field of view, and distance (which affects resolution) are considered in determining what positions can actually view a target. Resolution and distortion can be controlled by constraints on view angle. The camera is configured such that any target within the expected range of distances can be brought into focus. As with sensor parameters, reducing the number of feature detectability variables explicitly considered during planning reduces the dimensionality of the problem.

View planning methods can be broadly categorized as model-based and non-model-based [4]. Model-based methods assume some a priori model of the object or feature to be modeled, while non-model based methods do not. For planetary missions, operators collect as much information about the mission target as they can before landing, so it makes sense to expect this information for the pit modeling case and to exploit it to build a coarse a priori model.

In the photometric stereo work of Sakane and Sato, camera and illumination positions are planned to reconstruct surface normals for an object with known coarse position but some uncertainty in position and orientation [5]. They use a Gaussian sphere to predict surface normals and compute estimated reliability and detectability metrics. They use a generate and test method with potential camera and illumination positions on a geodesic dome, compute these metrics for each, and select the best positions. For a similar problem, Solomon and Ikeuchi determine an exact cover of each face (described by its estimated normal) by a set of light sources placed in an icosahedron centered on the object [6], [7], [8]. Camera position is chosen from a set of possible viewpoints (a generate and test method) to minimize the foreshortening of any faces visible in that view. Additional views are added until all faces to be inspected are covered.

Cowan and Bergman model polygonal objects with a lower bound on resolution by using a synthesis method in which camera positions lie on the intersection of spheres (with radius determined by the maximum distance for a given resolution) centered at vertices on the polygon's convex hull [9]. They compute a region of acceptable light source positions, using limits on minimum and maximum distance to get a goal intensity of reflected light, and subtracting out regions that would result in the camera seeing specular reflections. In planning for pit modeling, the distance between the light source (the sun) and the target is not controllable, but a range of illumination angles are available, depending on time of day, so a "region of acceptable light source placement" is handled by absolute constraints on illumination angle for imaging a particular patch of pit terrain, which in turn determines the times when the patch can be imaged.

Tarabanis, Allen and Tsai use a synthesis method for determining camera position, orientation and imaging parameters that computes regions in 8-dimensional space in which target features are visible, then optimizes over this space to find the point farthest from the bounds of the region [3]. Choosing such points that are far from the edges that separate acceptable from unacceptable views makes the method more robust to uncertainty. Robustness to uncertainty could be very important for pit modeling. It is addressed in pre-processing steps by the choice of potential rover positions and views, including the constraint that target terrain patches are smaller than what a camera could cover in one view. It will not be explicitly considered in planning.

Scott, Roth, and Rivest categorize model-based methods for 3D object reconstruction into those using set theory, graph theory, or computational geometry[4]. Set theory methods use a visibility matrix to encode which surface features on the target are visible from which viewpoints. Graph theory methods use aspect graphs, where aspect is defined as the set of viewpoints that have qualitatively the same view, and arcs connect adjacent aspects in the graph. Computational geometry methods use the art gallery problem, which looks for the minimum number and placement of guards to see all internal walls of an art gallery. In this work, visibility matrices are used to track which terrain patches are visible from which rover positions.

In addition to model-based work, non-model-based methods have also been tried for 3D reconstruction. These methods often take a "next-best-view" approach. Given what has been seen so far, the approach generates the next-best-view that will provide the most new information. In some cases, overlap with existing data to facilitate model building is also considered [10]. Overlap between views is important for pit modeling if one coherent model is desired instead of a set of discrete, 3D patches. Overlap is addressed in this work by setting target terrain patch sizes smaller than the camera's view, but it is not explicitly considered in planning.

How the 3D view target is represented also matters. This could be based on volume, surface area, or small features of specific interest. Methods that model polyhedral objects often use target vertices and edges in view planning, and assume a constant normal vector for faces. For more irregular targets, such as planetary pits, the volume or surface area that views observe could be discretized finely or coarsely into voxels or surface area patches. Kruse, Gutsche and Wahl present a volumetric method for planning sensor views to explore a previously unknown 3D space where each voxel in a 3D grid is marked as either occupied, free or unknown [11]. An early effort in pit modeling by the authors used a similar voxel-based approach with occupied, free or unknown voxels

[12]. Given the large size of planetary pits, any fine voxel discretization will quickly become very expensive in terms of memory and computation, and voxelizations smaller than about 0.5m/pixel were found to be infeasible in that work. For the work presented in this paper, a surface-based method is used to represent terrain shape.

For most work in planning viewpoints and illumination, the illumination is either considered completely out of the planner's control and generally not important to the problem, or completely under the planner's control, meaning that any of a set of illumination settings can be selected at any time [9], [3], [6], [7], [8]. This differs from view trajectory planning, where there are a set of potential illumination conditions, but they happen at specific times. This means that the planner can select illumination conditions that are advantageous to the model-building objective, but it cannot necessarily capture views of all target patches under the same illumination condition, since this illumination condition will not last indefinitely. The work presented in this paper considers view planning with time-dependant illumination. Another major difference between traditional view planning methods that do not consider time and view trajectory planning is the fact that for traditional view planning, it is possible to get full coverage of potentially viewable areas, while for view trajectory planning it may be impossible to do so before a time limit (e.g. the end of a rover's mission due to onset of lunar night or Martian winter).

Much early work in view planning centered around factory inspection and modeling tasks. While Kruse, Gutsche and Wahl used a metric of distance traveled between the current sensor configuration and the next view, most classical view planning methods do not. The most optimal views can be selected no matter what positions they are captured from, and the order can be determined based on metrics like information gain. For a mobile robot, choosing the next view from an information-optimal perspective without consideration for distance and navigation contingencies may produce unreasonably long paths. For view trajectory planning for pit modeling, distance traveled between viewpoints must be considered.

View Planning from Mobile Robots

Next best view has been applied to the robotic exploration of unknown environments [13]. Sawhney, Krishna and Srinathan use the amount of visible (but yet unseen) terrain combined with distance to determine the next best view for individuals in a multi-robot team. They find that the metric computed as (amount of unseen terrain)/distance is the most successful out of several evaluated [14]. This method was tried for pit modeling in the authors' earlier work, but it produced long path lengths for a single rover [12].

Moorehead describes robotic exploration as a Prize-Collecting Traveling Salesman problem [15]. The salesman gets a prize for visiting each city, but incurs a cost for travel between cities, and prizes in each city are independent. However, prizes in his work are the expected information gain at each location, and thus they change as the robot travels and information is gathered, so the prize in one city is not truly independent from the prizes in previously visited cities. This is also the situation in planning for pit modeling. Two positions may have views of multiple patches of terrain, with some overlap. Once one of these positions has been visited and the views visible from that position have been taken, the other will not have views to as many unseen patches.

Model-based view planning methods have also been applied to mobile robots. Hollinger et al. use uncertainty to plan sensor views for a ship inspection robot [16], [17]. They use a Gaussian process to model the surface of the ship hull. In their case, the cost of viewing is higher than the cost of moving between viewpoints, which is not the case with pit modeling. Englot and Hover address the same ship inspection task [18]. They address view planning as a coverage sampling problem and then address the multi-goal planning problem once the set of views are decided. Robot view configurations are randomly sampled, and they seek to build a feasible covering set for the modeled target from among the set of sampled views such that each geometric primitive is observed a requisite number of times. For the multi-goal planning problem, they use a Lazy Traveling Salesman Problem (TSP) solver on the assumption that the cost of planning point-to-point routes is high relative to the cost of solving the TSP, since there are relatively few viewpoints (100-200). This may also be the case for view trajectory planning for pit modeling.

Estlin et al. address the rover mission-activity planning problem: choosing which rocks a robot will view from a set of interesting scientific targets, and in what order [19]. Science targets are selected by a decision layer and then ordered using a TSP heuristic solver. A global path planner provides distance estimates, to get from point to point. These distance estimates are used by the TSP solver.

Smith also addresses the rover mission activity planning problem [20]. He describes it as an over-subscription planning problem, where there are too many goals for the time or resources available. Planning for pit modeling is somewhat similar, in that there are far more positions with views of terrain than the rover should visit, though this is partly because positions will have overlapping sets of visible targets. Even if the rover covers all target patches, which may not be possible depending on day length and rover resources, it would not have visited every potential position to do so. While Estlin et al. assume that once the planner decides which targets will be viewed (and which will not), the set of activities is fixed and it can be handed over to a TSP solver [19], the order in which the rover chooses to sample the rocks will determine the distance it has to travel to get to each rock, and that determines the cost for each investigation activity. So, as Smith notes, the problem is more properly described as an Orienteering Problem instead of TSP. His method solves the Orienteering Problem as an intermediate step in planning. This enables the determination of path to impact the plan of which rocks should be visited. Pedersen, et al. use the same Orienteering Problem approach as Smith for rover mission activity planning [21].

Methods that combine view and path planning, as discussed above, generally do not take time into account, except to minimize the total time required to complete the task. For pit modeling, minimizing total task time is less important than ensuring that the time at which each view is captured provides acceptable illumination conditions.

The Orienteering Problem and Extensions

The Orienteering Problem (OP) is similar to the Traveling Salesman Problem, but in this case the agent does not have to visit every goal position, and each goal position provides some reward [22]. The agent seeks to maximize reward while minimizing path length.

Vansteenwegen, Souffriau, and Oudheusden provide a survey of approaches to the OP and several of its extensions [23].

In the formulation discussed by Vansteenwegen et al., the starting and ending nodes for the orienteering problem are fixed. Each node has an associated reward, and each edge has an associated travel time. The goal is to maximize reward within a time budget, T_{max}. Methods for solving the Orienteering Problem often involve finding a feasible path, and then doing a local search to improve the feasible path. An extension of the OP is the Team Orienteering Problem (TOP), where instead of a single agent there are multiple agents and each one has a route [23].

While the OP is similar to the view trajectory planning for pit modeling, the view trajectory planning problem also has time constraints on when goal positions can be visited to achieve rewards. This is more similar to the Orienteering Problem with Time Windows (OPTW), in which each node can only be visited during a certain time window. Methods to solve the OP do not always translate well to the OPTW [23]. In particular, swapping nodes with other nodes is complicated by the fact that they may not have equivalent time windows. Karbowska-Chilinska and Zabielski use a genetic algorithm to solve the OPTW [24].

Planning for pit modeling could be formulated as an OPTW, with nodes being a combination of position and coarse time divisions, and the time window being defined over the coarse time division. The value of each node would then be dependent on whether the illumination was good on the patches visible from a node's position at that node's time. However, the OPTW does not reflect the fact that the value of a node can change based on what nodes were visited previously, which can happen when visits to previous nodes have collected all the images necessary to model a particular terrain patch.

The Team Orienteering Problem with Time Windows (TOPTW) combines the TOP with the OPTW. Since this pit modeling work only considers a single rover, the team extensions of OP are not relevant. The TOPTW has also been applied to solve multi-day OPTW problems, assuming that the agent must start and end from pre-determined locations each day. This makes sense when considering the agent as a tourist, salesman, or delivery man. He would have to return to a hotel or to the depot each day. It makes less sense in the case of a planetary rover which will spend the night wherever it happens to be when night falls and continue the next day where it left off.

Mennell introduces the Sequence Dependent Team Orienteering Problem (SDTOP) [25]. In this problem, the value of a route depends on the order in which nodes were visited. For example, if node 5 is visited before node 7, a value of 10 would be added to the score, but if node 7 were visited before node 5, a value of 15 would be added. Planning for pit modeling differs from the SDTOP in that, while absolute time matters, the sequence in which two nodes were visited does not affect the overall value, though it may affect the total distance traveled.

The Target Visitation Problem (TVP) was introduced by Grundel and Jeffcoat, and further analyzed by Arulselvan, Commander, and Pardalos [26], [27]. In the TVP, nodes have different values that is somewhat time dependent. The example given is a surveillance task for an aerial vehicle where intelligence suggests that a terrorist may be in a set of different locations, but there are different probabilities of him being in each location. Thus, the value is greater if the high-probability locations are visited first, but all locations should be checked, and there is a desire to minimize distance traveled. Arulselvan, Commander, and Pardalos use a hybrid genetic algorithm to solve the TVP. Like for the SDTOP, ordering matters in the TVP, where it does not in planning for pit modeling.

In the Multi Constrained Team Orienteering Problem with Time Windows, visits to each node also have specified durations and entry fees. Nodes are also sorted into categories, and once a given number of nodes in a category have been visited, additional nodes in that category are not valuable. Sylejmani, Dorn, and Musliu use the Tabu Search metaheuristic to solve the MCTOPTW [28]. Tabu search uses a number of moves that can potentially improve the path. "Tabu" memory keeps track of moves that cannot be performed for a certain number of iterations, which tends to diversify the search. They also use a technique of fast constraint checking to ensure that constraints (time windows, nodes per category, etc.) are not violated. In view trajectory planning for pit modeling, there is a desired number of views for each patch. The problem could potentially be formulated as an MCTOPTW, but this would require that each position-time-view combination be a separate node, which may not be desirable. Not only would this greatly increase the number of nodes in the problem, but if a rover is at a given position, it makes sense to capture all the views associated with that position (or at least all the views of unseen patches or those that meet constraints with prior views of a patch), and the MCTOPTW does not capture this connection between patches viewable from the same position. The MCTOPTW also does not handle the idea of relative constraints on angles between nodes.

In the generalized orienteering problem (GOP), the total score is a nonlinear function of the vertices visited [29], [30]. Specifically:

$$Z = \sum_{g=1}^{m} W_g \left[\{ \sum_{i \in P} \left[S_g\left(i\right) \right]^k \}^{\frac{1}{k}} \right] \tag{1}$$

This function sums over all nodes in the path P. Each node has a set of scores S_g in different categories. The example given is a tourist planning a trip who wants to see attractions that each have some score for natural beauty, cultural heritage, shopping, etc. Given that the tourist has seen several attractions with high scores in natural beauty, additional attractions with high natural beauty may not be as interesting anymore. Each tourist may have his or her own weightings, W_g for each type of score. This function tends to level off in score for each category as the number of nodes visited increases with high scores in that category increases. Wang, Golden and Wasil use a genetic algorithm to solve the GOP [29].

Planning view trajectories for pit modeling is similar to the GOP. Given position-time nodes with scores determined by the patches visible from that location, if a patch has already been viewed many times, it should not increase the node's score. However, there is an additional level of dependency between nodes in planning view trajectories for pit modeling. If the goal is to get stereo depth reconstruction for each patch, then there are two views required for each patch. After the first view of a patch, the second view would be of value if it meets the relative view angle constraints for stereo or no value if it did not. The GOP also does not include time windows. This work thus seeks to extend the GOP and the MCTOPTW and define a new problem, the Orienteering

Problem with Time Windows and Inter-Node Dependencies (OPTWIND), that handles inter-node dependencies such as relative constraints on view or illumination angle for a patch viewable from multiple nodes.

Model Building

To plan for model building, it is important to understand how illumination and view angles affect model quality. This work looks at stereo reconstruction from two or more images. For stereo reconstruction, very similar view angles to a target will produce poor-quality depth information, but it may be difficult to match features between images from very different view angles. So for stereo, there are both minimum and maximum constraints on relative view angle to get a good quality reconstruction. Matching features between images can also fail if the illumination angles in the two images are too different.

Stereo reconstruction has been done from orbit in planetary missions. In this case, the difference in view needed for stereo is achieved by moving and tilting the spacecraft. One stereo reconstruction method for orbital images is the open-source Ames Stereo Pipeline [31], [32]. Stereo from orbit is similar to the consideration of stereo in this pit modeling work, and can be contrasted against parallel stereo that achieves a difference in view based on camera separation perpendicular to the view direction. The larger the distance is between the cameras and the view target, the wider this separation, or baseline, must be. Stereo between multiple rover positions is used for pit modeling because of the cross-pit viewing distance.

The modeling of Victoria Crater by the MER Opportunity is an example of how planetary rovers currently model terrain features [33], [34]. Waypoints for Opportunity were chosen by operators on Earth instead of being planned autonomously. Both MER and MSL carry multiple stereo pairs of cameras, so stereo (over a short baseline) is commonly used. Bundle adjustment is also used for these rovers, to register image sets over the course of a rover trajectory [35].

Depending on the desired spatial resolution of the final model, a pit model could be composed of many images. An effective pipeline for modeling large features with large numbers of images on Earth consists of open source Bundler [36], [37] for bundle adjustment, and CMVS/PMVS multi-view stereo software [38], [39]. Their work has been demonstrated in reconstruction of tourist destinations like the Colosseum in Rome and the Piazza San Marco in Venice using thousands of tourist photos. This and several other sparse reconstruction (structure from motion or SFM) and dense reconstruction methods will be examined in Section 7.

3. Formulating the OPTWIND Problem

The formulation of the Orienteering Problem with Time Windows and Inter-Node Dependencies (OPTWIND) starts from the standard formulation of the Orienteering Problem with Time Windows (OPTW) [23]. The variable $x_{ij} = 1$ if the agent visits node n_i immediately before node n_j. The distance and time between nodes n_i and n_j are d_{ij} and τ_{ij}, respectively. For each node n_i, σ_i is the start of the visit to that node. M is a large constant. The time window for n_i is the interval $[O_i, C_i]$.

$$\sum_{j=2}^{N} x_{1j} = \sum_{i=1}^{N-1} x_{iN} = 1 \tag{2}$$

$$\sum_{i=1}^{N-1} x_{ik} = \sum_{j=2}^{N} x_{kj} \leq 1 \tag{3}$$

$$\sum_{i=1}^{N-1} \sum_{j=2}^{N} d_{ij} x_{ij} \leq D_{max} \tag{4}$$

$$\sigma_i + \tau_{ij} - \sigma_j \leq M\left(1 - x_{ij}\right); \forall i, j = 1, ..., N \tag{5}$$

$$O_i \leq \sigma_i \tag{6}$$

$$\sigma_i \leq C_i \tag{7}$$

Constraints in eq. 2 ensure that the path starts and ends at predetermined start and end locations. Constraints in eq. 3 ensure that the path is continous, and each node is visited no more than once. Eq. 4 ensures that the path does not exceed a maximum distance. In some work with the orienteering problem, this is expressed as a maximum time instead of a maximum distance, but time is also constrained by absolute time windows, and if the distance that could be traveled in between the beginning of the first time window and the end of the last is greater than the desired maximum distance, then distance should be constrained independently. Eq. 5 ensures that the timeline is consistent; in other words, if the agent travels from node n_i to node n_j, then arrival time at n_j cannot be less than the arrival time at n_i plus the travel time between n_i and n_j. The indicator y_i is also added, where $y_i = 1$ if node i is in the path, and zero otherwise.

In OPTW, each node has a fixed score, and the objective is to optimize the sum of the scores over all nodes in the path. Similar to the Generalized Orienteering Problem (GOP), each node has not just one score in the OPTWIND formulation, but scores in several categories, and the total score for the path is a function of the per-category scores for each node.

In OPTWIND, the score for each node n_i is the sum of the scores across all categories, $\{p_1, ..., p_k\}$. In each category, p_k, there are one or more ratings $\{\beta_{ik1}, ..., \beta_{ikz}, ..., \beta_{ikZ}\}$, where each β_{ikz} is an integer. The score in a given category for a given node is dependent on whether the ratings for that category meet absolute constraints for ratings and relative constraints with ratings in the same category for nodes previously visited. If constraints are met, the score for the node in that category is S_k, up to a total of $S_{k,max}$ for that category across all nodes.

The total value for the route that the problem seeks to optimize is then:

$$\sum_{k=1}^{K} min\{S_{k,max}, \sum_{i=1}^{N} [S_k a\left(n_i, p_k\right)\{f\left(\{n_1, ..., n_i\}, p_k\right) + r\left(\{n_1, ..., n_i\}, p_k\right)\}y_i]\} \tag{8}$$

The expression

$$a\left(n_i, p_k\right)\left\{f\left(\{n_1, ..., n_i\}, p_k\right)+r\left(\{n_1, ..., n_i\}, p_k\right)\right\} \quad (9)$$

evaluates to a binary value that indicates whether or not the score S_k can be earned. The function a indicates whether the absolute rating constraints are met. The function r indicates whether two or more observations of a category meet relative rating constraints. The absolute and relative constraint functions a and r are defined using absolute thresholds $A_{z,min}$ and $A_{z,max}$, and relative thresholds $R_{z,min}$ and $R_{z,max}$ for each type of rating.

$$a\left(n_i, p_k\right)=\prod_{z=1}^{Z}\left[min\left[max\left(\beta_{ikz}-A_{z,min}, 0\right), 1\right] * min\left[-min\left(\beta_{ikz}-A_{z,max}, 0\right), 1\right]\right] \quad (10)$$

$$r\left(\{n_1, ..., n_i\}, p_k\right)=max\Big\{\sum_{j=1}^{i-1}\Big[y_j * \prod_{z=1}^{Z}\Big(min\left[max\left(|\beta_{ikz}-\beta_{jkz}|-R_{z,min}, 0\right), 1\right] * min\left[-min\left(|\beta_{ikz}-\beta_{jkz}|-R_{z,max}, 0\right), 1\right]\Big)\Big], 1\Big\} \quad (11)$$

The function f indicates whether this is the first node on the path for which p_k meets absolute rating constraints.

$$f\left(\{n_1, ..., n_i\}, p_k\right)= min\Big\{-min\left[\sum_{j=1}^{i}\left(a\left(n_i, p_k\right)\right)-1, 0\right], 1\Big\} \quad (12)$$

View Trajectory Planning for Pit Modeling as an OPTWIND

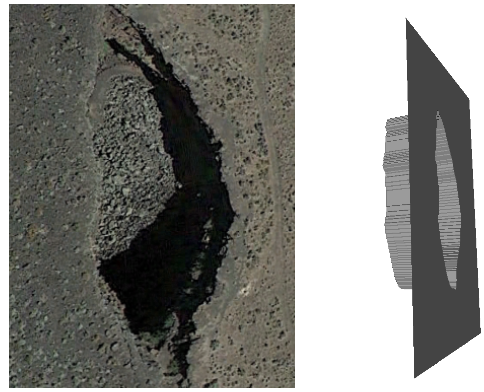
To formulate view trajectory planning for pit modeling as an OPTWIND, one node is created for each pair (t_i, l_i). The time windows are calculated as:

$$O_i = t_i T \text{ and } C_i = \left(t_i+1\right) T \quad (13)$$

where T is the length of a coarse time division. One category is defined for each target patch. The rating β_1 for each patch is the view angle from the rover position l_i in node n_i, and the rating β_2 is the illumination angle at time t_i in node n_i. Additional β values could be added to represent vertical view angles. Distances d_{ij} between nodes n_i and n_j are computed as $d(l_i, l_j)$, and times τ_{ij} are computed from d_{ij} using the rover speed. This application also relaxes the constraint in eq. 2 to only specify a start position and not an end position.

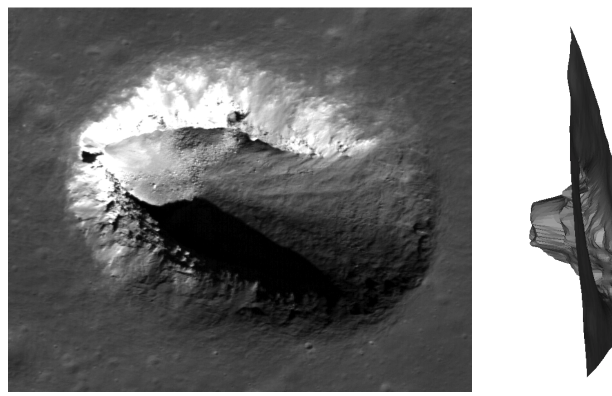
4. INPUTS

Prior models of pit geometry can be as simple as cylinders with a given radius and height. A cylinder crudely represents pit walls, and pit floor may or may not be considered. Pit diameter can be measured from satellite imagery, and pit depth can be estimated from the length of shadows. Many natural pits of ultimate interest here are highly irregular. For these irregular pits, the outline of the pit can be projected down vertically to the estimated pit depth. Fig. 3 illustrates this concept. The pit was viewed from above using Google Earth, and the outline of the pit was traced and saved as latitude and longitude values in a KML file [40]. The outline was then projected down to the estimated depth of the pit to create a prior model.



(a) Satellite view (b) Prior model

Figure 3. Prior model example for projection of outline, using King's Bowl pit. (Left image from [41], right image created by the authors)



(a) Satellite view (b) Prior model

Figure 4. Prior model example for satellite stereo reconstruction, using a pit in the Moon's Lacus Mortis region. (Left image from [42], right image created by the authors.)

If multiple satellite images (at the right viewing angles) are available, then a stereo model can be constructed from a pair of images. Fig. 4 illustrates this concept. Images from the Lunar Reconnaissance Orbiter's Narrow Angle Camera of a target pit in the Moon's Lacus Mortis region were used with Ames Stereo Pipeline [31], [32] to create a stereo model. Because this pit has a ramp on one side, stereo will be much more effective than the projection of outline method in representing its shape.

Time information used in view trajectory planning consists of the length of daylight - for the Moon this is 14.77 Earth days, or about 354 hours - as well as start and end times for the modeling mission, and a desired number of divisions to consider for the mission period. This last is based on the insight that, while illumination changes continuously with time, over the time scales required to take a single view, which may be less than a second up to a minute or more for high dynamic range imaging, illumination does not change that much. The number of time divisions should be set such that within each time division, the illumination can be considered essentially the same for purposes of planning. This could mean that the illumination angle on a patch of terrain changes less than some threshold (5 degrees, 20 degrees, etc.), or that the set of illuminated patches does not change more than some threshold, which in turn depends on patch size.

The illumination trajectory is computed from time information and a given latitude. The sun is assumed to travel 360 degrees relative to the pit in one day length, in a plane that is latitude degrees from vertical. One light vector is computed for the center of each time division.

Restrictions on rover travel include a maximum distance traveled in a single time division and a minimum distance that a rover must maintain from the pit edge. The upper bound on maximum distance traveled is set by the maximum rover speed, but the value is set much lower for this work to account for other un-modeled mission considerations.

5. Input Processing

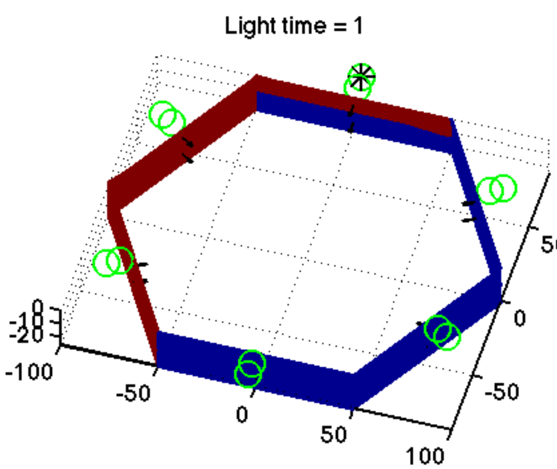


Figure 5. Coarse pit discretization with 2 vertical and 6 angular patches. Potential rover positions are shown as green circles.

A coarse prior model of a pit can be divided into a set of surface patches, where the normal of a surface patch is determined from a plane fit to the coarse model points in the local neighborhood of the patch. The patch normal can then be used to determine the view angle to the patch from various positions and the illumination angle on the patch at various times. Fig. 5 shows an example patch tiling for a simple cylindrical pit with no visible floor. Fig. 6 shows an example tiling for the King's Bowl pit. The size of the patches could be fixed, or it could vary over the pit surface. In this work, a fixed patch size was used. The size of the patch should be smaller than the footprint of a camera view to ensure that the camera view can cover the patch.

For any pit-modeling task, there will be a start and stop date/time. The sun direction can then be computed from

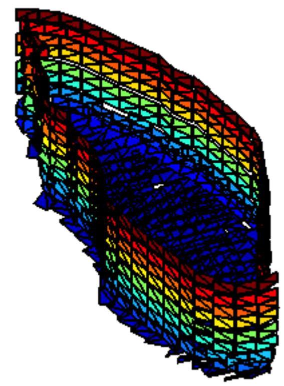


Figure 6. A tiling of the coarse model of King's Bowl pit into surface patches

ephemeris data. This is done using SPICE [43]. For simplicity, it is assumed that the sun is a directional light source. This sun direction and the patch normals can be used to determine patch illumination angles, and patch positions combined with a coarse prior model of the pit can be used to do ray-tracing to determine shadowed patches [44]. Fig. 7 shows an example of a lighting computation for a cylindrical pit model.

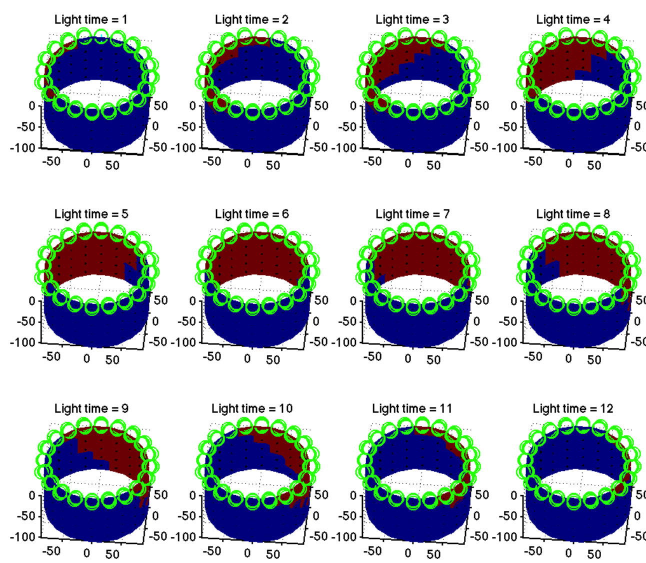


Figure 7. Simple geometric with 5 vertical and 20 angular divisions and 12 time divisions, showing a change in lighting over time. Red indicates that a patch is lit, and blue indicates that it is not lit.

A set of positions for the rover to consider while planning, and a set of path lengths between these positions are generated. These path lengths could be straight-line distances, or, given a digital elevation map (DEM), paths could be planned automatically, avoiding high slopes and obstacles. For the simple pit models in Fig. 5 and Fig. 7, the green circles represent potential rover positions.

For King's Bowl, positions were generated at roughly 8 m increments along both the east and west side of the pit (see Fig. 8). Because a digital elevation map was not part of the prior model, paths between adjacent positions were manually estimated (avoiding cracks and holes visible in overhead

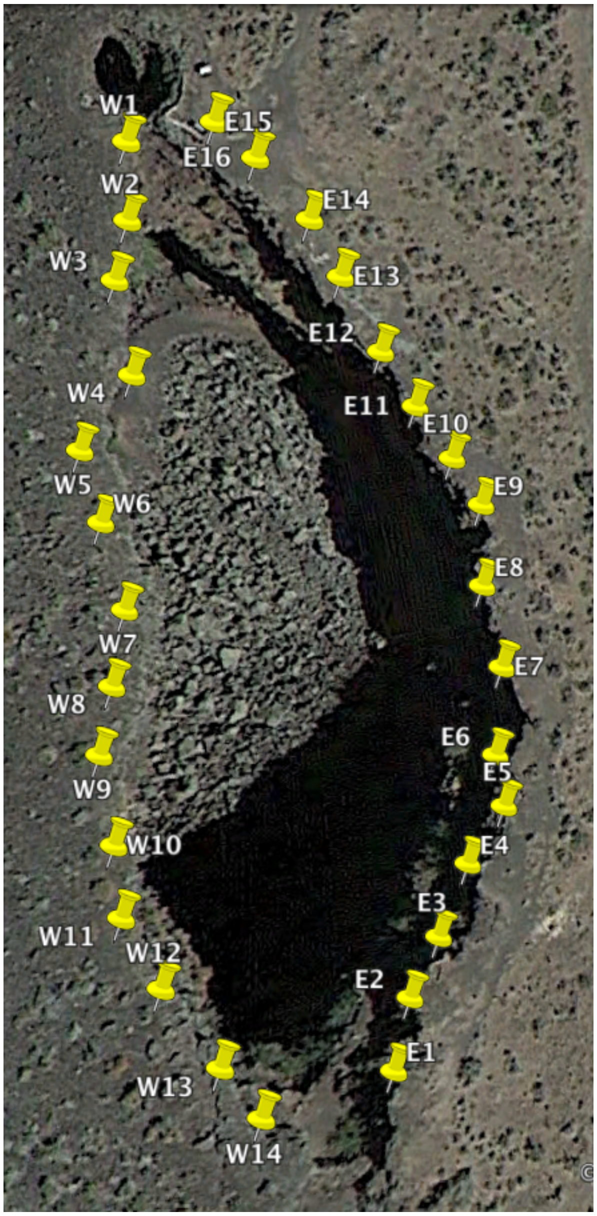


Figure 8. Overhead view of Kingsbowl pit. Potential positions are marked by yellow push-pins. (View from Google Earth [40]. Base image from [41])

imagery) and measured using Google Earth [40]. Paths are assumed to be bi-directional. Dijksra's algorithm is run on a graph with position nodes and path-length edges to compute the distance from each position to each other position [45].

To determine the visibility of patches from various rover positions, one can either generate rover positions and views and test which patch(es) they can see (generate and test) or determine for each patch which rover positions can see it (synthesis). Although synthesis methods for visibility computation are an intriguing option for view trajectory planning, for the King's Bowl pit experiments, camera positions were chosen from among a set of positions already marked from a prior experiment (see Fig. 8, so the generate and test method was used.

To determine if patches are visible in a given view from a rover position, ray tracing is done from the camera position [44]. The number of rays is selected based on the size of a patch and the spread between adjacent rays at the maximum expected viewing distance. The distance between ray tips at that distance should be somewhat less than the patch size. For patches that are intersected by rays, the four vectors between the camera center and the patch corners are checked against the camera field of view to ensure that the entire patch can be seen. Two dimensional view angles, computed by comparing the direction of vectors from the camera to the patch centers with the patch normals, are stored for each valid position-patch pairing.

6. View Trajectory Planning

A view trajectory plan consists of a sequence of times, rover postions, and patches to be viewed. When the end-goal of planning is to have a high-quality model of a pit, plans are evaluated by the quality of the resulting model. The quality of the resulting model cannot be evaluated without first executing the plan, capturing the images, and building the model, so when comparing potential plans, an estimate of model quality is used.

The estimate of model quality is based on lighting and view angle constraints. For an image of a patch to add value to the model, it must be lit within the illumination angle threshold. An absolute view angle threshold is imposed, and images at more oblique angles do not add value. If stereo is the assumed method of 3D reconstruction, both minimum and maximum constraints on relative view angle are used. For each patch, a score of 0.5 is given for a single image that meets constraints. A score of 1 is given if there is a pair of views that meet both absolute and relative constraints. The sum of per-patch scores is computed, and the model value is the percentage of the maximum possible score that this sum achieves. This takes into account that some patches may never be visible, or may never be sufficiently lit, reducing the maximum possible score in these cases.

Two algorithms for solving the OPTWIND are used in experiments. One takes a greedy approach and takes any views that meet absolute and relative view and lighting angle constraints. The other is the timewindow algorithm. Before this algorithm starts, it is assumed that time has been coarsely divided, based on significant changes in availability or value of positions, similar to the start time, end time, and time divisions. The algorithm first computes a value for each position at each time, as the sum of potential values for each target type available from that position, weighted by one over the total length of time during which that target type is available from any position. Positions within each coarse time division are then ordered by this value. Starting with the top position for each time division, the algorithm tries to find a feasible trajectory that includes one position per time division and meets distance constraints, proceeding to lower-valued positions for a given time if necessary. From the feasible trajectory, it tries to add positions, ordered by the difference in the set of targets available (the larger the difference, the better), within each coarse time division and remaining within the distance constraints. It does a local search to reduce the trajectory distance. It then determines which targets do not have the desired number of views, and tries to add positions, ordered by the number of target types they meet constraints for, and thus improve value. The insert by set difference, reduce trajectory distance, and insert to satisfy constraints steps are repeated until the change in

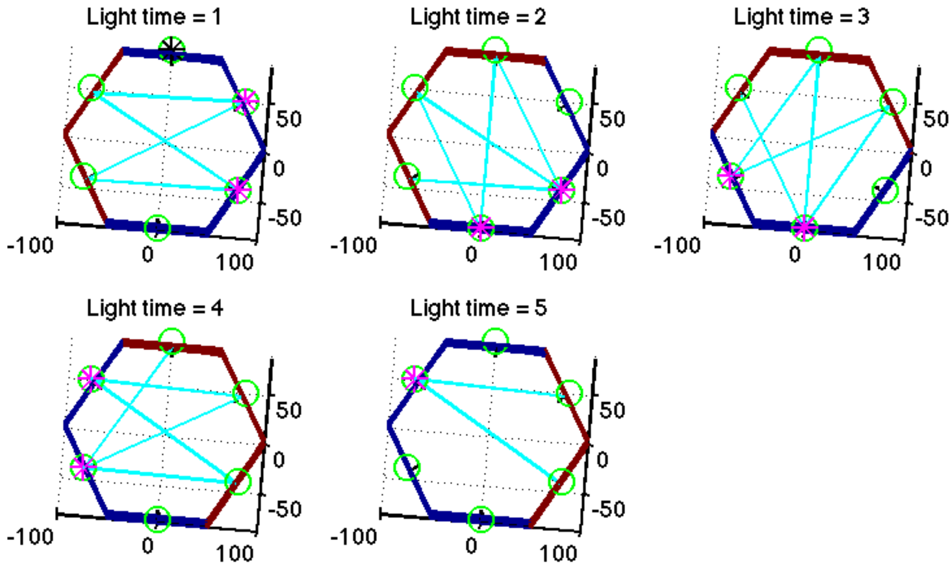


Figure 9. Example of a planned view trajectory for a simple cylindrical pit. Black asterisk indicates starting position. Magenta asterisks indicate rover positions, and cyan lines indicate views.

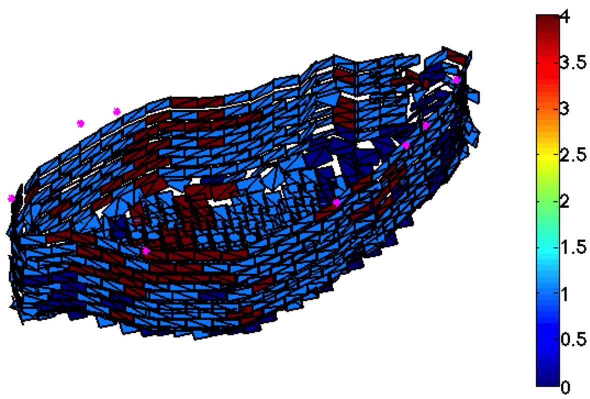


Figure 10. Patches viewed under the greedy plan. Patches are colored by number of views. Magenta asterisks indicate the rover positions in this plan.

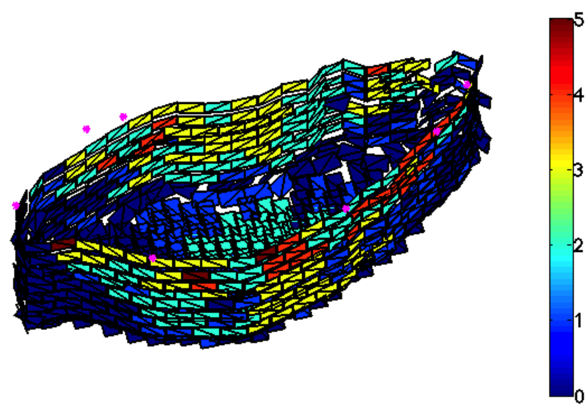


Figure 11. Patches viewed under the timewindow plan. Patches are colored by number of views. Magenta asterisks indicate the rover positions in this plan.

paths is lower than some specified threshold, or a specified maximum number of iterations is reached.

Patches viewed in the view trajectory plans for both the greedy and the timewindow method are shown in Fig. 10 and Fig. 11.

7. Model Building

Model building takes images captured according to a view trajectory plan and combines them into a detailed model. First, the relative positions of the cameras are determined in a structure-from-motion step, and then a dense reconstruction is computed given those positions. Here we evaluate several existing modeling methods, a combination of structure-from-motion and dense reconstruction algorithms, to determine which is most effective on data from an analog field site.

Camera reconstructions of the King's Bowl pit were evaluated using a set of images taken at positions E14 at time 17:19 and E13 at time 16:11 (shown in Fig. 8). Images were taken with a Canon EOS Rebel T3 DSLR camera with a 50mm lens. Reconstructed models from images were compared to a ground truth LiDAR model of the site.

The pipelines evaluated were the Multi-View Environment (MVE) in [46], VisualSFM [47], [48] for structure-from-motion and MVE for dense reconstruction, VisualSFM for structure-from-motion and PMVS/CMVS [49], [50] for dense reconstruction, OpenMVG [51] for structure-from-motion and MVE for dense reconstruction, Bundler [52], [53], Bundler for structure-from-motion and PMVS/CMVS for dense reconstruction, and Bundler for structure-from-motion and MVE for dense reconstruction. The combination of approaches can be seen in Table 1.

OpenMVG implemented two methods for computing global rotation from a list of relative estimates. The first method employed Martinec's algorithm detailed in [54] and the second method employed an algorithm developed by Chatterjee, et. al. detailed in [55]. OpenMVG1 in Table 1 refers to the first method and OpenMVG2 refers to the second method.

Fuhrman et. al. developed the Multi-View Environment (MVE) in [46] that takes images as input and outputs a dense point cloud. Their algorithm includes structure-from-motion, multi-view stereo reconstruction, and dense point cloud generation from multi-scale data. The structure-from-motion and dense point cloud generation are run together on the input data and compared to other structure-motion-algorithms combined with the MVE dense point cloud generation. The full reconstruction pipline can be run using the MVE software. Alternatively, MVE can take as input sparse reconstruction from another structure-from-motion software and output a dense reconstruction.

The Clustering Views for Multi-View Stereo (CMVS) [50] and Patch-based Multi-view Stereo [49] algorithms of Furukawa, et. al. were also evaluated to generate dense point clouds from the camera images. The CMVS algorithm produces a dense set of rectangular patches that cover the surface being rceconstructed by enforcing local photometric consistency and global visibility constraints. Each image is associated with a regular grid of cells and, if possible, at least one patch p in every cell is reconstructed. The photometric consistency between a reference image $R(p)$ and the set of images $T(p)$ is measured by projecting a patch into the two images and computing the normalized cross correlation. The maximum score is used to compute an estimate of the position $c(p)$ and surface normal $n(p)$ using the following equation.

$$\tilde{N}(p) = \frac{1}{|T(p)|-1} \sum_{I \in T(p), I \neq R(p)} N(p, R(p), I)$$

Each of the methods reconstructed a camera model of the pit of varying quality. The quality of the model was measured by comparing to ground truth collected by a LiDAR. Each camera model was coarsely registered to the LiDAR model using handpicked points. Following the coarse registration, a fine registration was computed using the Iterative Closest Point (ICP) algorithm [56], [57], [58].

In order to accurately compare the LiDAR ground truth model to the camera models generated by the pipelines, the vertices in the models were voxelized using a density of 0.2 meters. The models could then be directly compared to determine the statistics in Table 1.

An analysis of the results concludes that the structure-from-motion component of Bundler or VisualSFM should be combined with the dense reconstruction of MVE. The other methods (i.e. OpenMVG, CMVS/PMVS, and the structure-from-motion component of MVE) produced significantly poorer results. However, it must be noted that these combinations of methods were run on a single dataset with fairly consistent lighting conditions. Better results may be achieved with these pipelines using different datasets and lighting conditions.

Table 1. Dense Reconstruction Pipelines

Structure from motion	Reconstruction	Coverage	% camera values not found in ground truth
MVE	MVE	5.3	75.7
VisualSFM	**MVE**	**20.8**	42.4
VisualSFM	CMVS/PMVS	3.8	23.5
OpenMVG1	MVE	2.9	92.4
OpenMVG2	MVE	2.9	91.4
Bundler	—	14.5	24.5
Bundler	CMVS/PMVS	3.6	22.9
Bundler	**MVE**	**22.4**	45.0

8. Results

Two field experiments in pit model reconstruction were executed. For the first, a "full set" of images was used. This means that for evenly spaced points around the pit rim, a set of all possible views was collected (spaced at approximately half field-of-view angle rotations in pan and tilt) from the left-most pan angle that viewed pit wall to the right-most pan angle that viewed pit wall. In the second experiment, a set of images from a planned view trajectory was used for model reconstruction.

Model Reconstruction: "Full View Set"

For a skylight pit in the Indian Tunnel lava tube cave, images were collected from tripod-mounted DSLR cameras. 205 images total were collected from 14 stations near the southern end of the Indian tunnel Skylight. Tripod stations were spaced with a target distance of roughly 4.5 meters. Stations were adjusted as needed to accomodate hazardous terrain, and precise measurements of station positions were not done. At each station, the tripod was manually leveled by eye using an integrated bubble level. The tripod was adjusted sch that the camera tilt was zero. The camera was panned to the left-most ege of the pit (in the current view). Additional views were taken by tilting down by 22.5 degrees to 45 degrees and panning by 30 degrees until hte right-most edge of the pit was reached.

Two Canon EOS Rebel T3 DSLR cameras with manually adjustable zoom lenses, 18mm focal length, and field of view of approximately 45 degrees vertical and 63 degrees horizontal were used to take images of the skylight. Each image was taken with auto-exposure using center-weighted average metering.

Other camera parameters were fixed for all images, including ISO 100, aperture F8 and white balance of 5200K (daylight).

A 3D model of the pit was constructed using Bundler for bundle adjustment and CMVS/PMVS2 for dense reconstruction. Figure 12 shows a view of the colorized point cloud created using this method. A mesh model is generated by applying a Poisson reconstruction.

Model Reconstruction: Planned View Trajectory

This experiment was conducted with King's Bowl data. A Canon EOS Rebel T3 DSLR camera with a 50mm fixed focal length lens was used. Because of the smaller field of view, and the larger size of the King's Bowl pit, the view trajectory planning did not have access to the "full set" as in the previous experiment.

Figure 12. Side view of colorized point cloud model created from dense reconstruction from CMVS/PMVS2. The skylight pit is in the center, and the segments of the lava tube cave can be seen extending on either side.

Figure 13. Mesh model of a skylight pit in Indan Tunnel cave created with Bundler, CMVS-PMVS, and Poisson Reconstruction

Fig. 14 shows the reconstruction for the greedy algorithm's view trajectory plan. Fig. 15 shows the reconstruction for the timewindow algorithm's view trajectory plan.

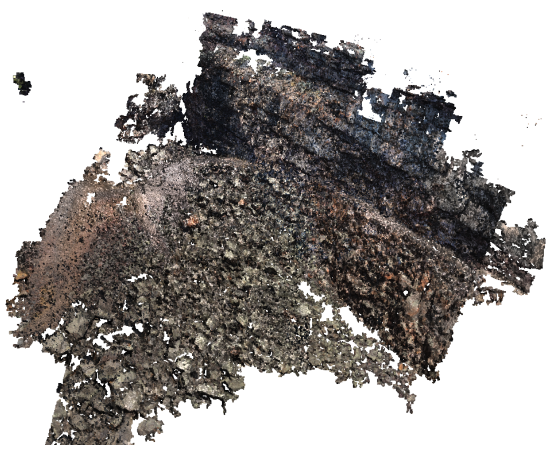


Figure 14. A 3d reconstruction from a planned set of images of King's Bowl, using the greedy planning method

9. Summary

This work has modeled pits in natural terrain on Earth that are analogs for planetary pits. Methods have been developed for converting a real-world pit modeling problem into a view trajectory planning problem and outputting planned image locations, directions, and times. This end-to-end pipeline has been demonstrated for the King's Bowl pit. This work presents preliminary results for a portion of a field dataset of pit images. Image-based reconstruction methods were

Figure 15. A 3d reconstruction from a planned set of images of King's Bowl, using the timewindow planning method

compared and promising and unpromising methods were identified.

10. Conclusions and Future Work

Preliminary results indicate that the combination of VisualSFM/MVE and Bundler/MVE for SFM/dense reconstruction are most promising for this pit modeling application due to the substantial model coverage they provide. The Bundler/CMVS-PMVS combination, while displaying low coverage on the comparison dataset, has produced good results on other field datasets for pits. One interesting point to note is that the Indian Tunnel dataset, while similar in size to the comparison dataset, contains more camera positions, and these positions are relatively evenly spaced around the pit to be modeled. In the long term, reconstruction of a patch using fewer images is advantageous in a planetary setting, due to severe limitations in on-board processing power and datarate to Earth. Combinations using OpenMVG for SFM did not prove at all promising for the pit modeling application due to the low coverage and the very high percentage of reconstructed points outside the range of the ground truth model.

This paper reports an early analysis of a part of a large dataset for King's Bowl; future work will perform further processing of this field data and may provide more comprehensive models for the pit. Future areas of investigation include a more in-depth analysis of parameters used for planning, such as absolute and relative view angle, and relative lighting angle.

This work looked at the pipeline for view trajectory planning, from inputs to model building. While two planning methods were applied, they were not analyzed in depth. Future work will more fully investigate the planning portion of the pipeline. This includes comparing methods over multiple test cases, examining a wider range of planning methods, (such as those from vehicle routing or AI planning and scheduling work), and investigating ways in which the 3D reconstruction of contiguous sets of patches is given higher value so that one large model can easily be produced, instead of many small ones.

ACKNOWLEDGMENTS

Support for this research comes from two NASA Space Technology Research Fellowships (NSTRF), grant numbers NNX12AM43H and NNX14AL66H, a NASA Innovative Advanced Concepts (NIAC) program grant number NNX12AQ55G, and the Paul and Daisy Soros Fellowship for New Americans. The authors would also like to thank the FINESSE SSERVI project for the chance to participate in a field campaign at Craters of the Moon National Monument and Preserve, Idaho. Craters of the Moon field team members Chuck Whittaker and Ander Solorzano played a critical role in data collection for this work. Craters of the Moon park geologist Doug Owen provided valuable advice for successful field operations. Uland Wong provided a merged and filtered LiDAR model for King's Bowl pit, used as ground truth in this work.

REFERENCES

[1] V. R. Oberbeck, W. L. Quaide, and R. Greeley, "On the origin of lunar sinuous rilles," *Modern Geology*, pp. 75–80, 1969.

[2] G. E. Cushing, "Candidate cave entrances on mars," *Journal of Cave and Karst Studies*, vol. 74, no. 1, pp. 33–47, 2012.

[3] K. A. Tarabanis, P. K. Allen, and R. Y. Tsai, "A Survev of Sensor Planning in Commter Vision," *Robotics and Automation, IEEE Transactions on*, vol. 11, no. 1, pp. 86–104, 1995.

[4] W. R. Scott, G. Roth, and J.-F. Rivest, "View Planning for Automated Three-Dimensional Object Reconstruction Inspection," *ACM Computing Surveys*, vol. 35, no. 1, pp. 64–96, 2003.

[5] S. Sakane and T. Sato, "Automatic Planning of Light Source and Camera Placement for an Active Photometric Stereo System," in *Proceedings 1991 IEEE International Conference on Robotics and Automation*, 1991, pp. 1080–1087.

[6] F. Solomon and K. Ikeuchi, "An illumination planner for Lambertian polyhedral objects," *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1719–1725, 1995. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=526016

[7] ——, "An Illumination Planner for Convex and Concave Lambertian Polyhedral Objects," Carnegie Mellon University, Pittsburgh, PA, Tech. Rep., 1995.

[8] F. Solomon, "Illumination planning for photometric measurements," PhD, Carnegie Mellon University, 1996. [Online]. Available: http://www-preview.ri.cmu.edu/pub_files/pub3/solomon_fredric_1996_1/solomon_fredric_1996_1.pdf

[9] C. Cowan and A. Bergman, "Determining the camera and light source location for a visual task," in *Proceedings, 1989 International Conference on Robotics and Automation*. IEEE Comput. Soc. Press, 1989, pp. 509–514. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=100037

[10] R. Pito, "A sensor-based solution to the next best view problem," in *Proc. Intl. Conf. on Pattern Recognition*, 1996.

[11] E. Kruse, R. Gutsche, and F. M. Wahl, "Efficient, Iterative, Sensor Based 3-D Map Building Using Rating Functions in Configuration Space," in *ICRA*, Minneapolis, Minnesota, 1996.

[12] H. Jones, U. Wong, K. Peterson, J. Koenig, A. Sheshadri, and W. R. L. Whittaker, "Complementary Flyover and Rover Sensing for Superior Modeling of Planetary Features," in *Proceedings of the 8th International Conference on Field and Service Robotics*. Springer Verlag, Jul. 2012.

[13] K. Nagatani, T. Matsuzawa, and K. Yoshida, "Scanpoint Planning and 3-D Map Building for a 3-D Laser Range Scanner in an Outdoor Environment," in *FSR*, 2009.

[14] R. Sawhney, K. M. Krishna, and K. Srinathan, "On fast exploration in 2D and 3D terrains with multiple robots," in *Proc. Intl. Conf. on Autonom. Agents and Multiagent Systems*, 2009.

[15] S. J. Moorehead, "Autonomous Surface Exploration for Mobile Robots," PhD, Carnegie Mellon University, 2001.

[16] G. A. Hollinger, B. Englot, F. Hover, U. Mitra, and G. S. Sukhatme, "Uncertainty-Driven View Planning for Underwater Inspection," in *ICRA*, 2012.

[17] G. a. Hollinger, B. Englot, F. S. Hover, U. Mitra, and G. S. Sukhatme, "Active planning for underwater inspection and the benefit of adaptivity," *The International Journal of Robotics Research*, vol. 32, no. 1, pp. 3–18, Nov. 2013. [Online]. Available: http://ijr.sagepub.com/cgi/doi/10.1177/0278364912467485

[18] B. Englot and F. S. Hover, "Three-dimensional coverage planning for an underwater inspection robot," *The International Journal of Robotics Research*, vol. 32, no. 9-10, pp. 1048–1073, Sep. 2013. [Online]. Available: http://ijr.sagepub.com/cgi/doi/10.1177/0278364913490046

[19] T. Estlin, F. Fisher, D. Gaines, C. Chouinard, S. Schaffer, and I. Nesnas, "Continuous Planning and Execution for an Autonomous Rover," in *3rd Intl. NASA Workshop on Planning and Scheduling for Space*, 2002.

[20] D. E. Smith, "Choosing Objectives in Over-Subscription Planning." in *Intl. Conf. on Automated Planning and Scheduling*, 2004. [Online]. Available: http://www.aaai.org/Papers/ICAPS/2004/ICAPS04-046.pdf

[21] L. Pedersen, D. E. Smith, M. Deans, R. Sargent, C. Kunz, D. Lees, and S. Rajagopalan, "Mission Planning and Target Tracking for Autonomous Instrument Placement," in *Aerospace Conference*, Big Sky, MT, 2005, pp. 34–51.

[22] C. Keller, "Algorithms to solve the orienteering problem: A comparison," *European Journal of Operational Research*, vol. 41, no. 2, pp. 224–231, Jul.

1989. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0377221789903883

[23] P. Vansteenwegen, W. Souffriau, and D. V. Oudheusden, "The orienteering problem: A survey," *European Journal of Operational Research*, vol. 209, no. 1, pp. 1–10, Feb. 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0377221710002973

[24] J. Karbowska-Chilinska and P. Zabielski, "Genetic Algorithm Solving the Orienteering Problem with Time Windows," in *Advances in Systems Science*, ser. Advances in Intelligent Systems and Computing, J. Switek, A. Grzech, P. Switek, and J. M. Tomczak, Eds. Cham: Springer International Publishing, 2014, vol. 240, pp. 609–619. [Online]. Available: http://link.springer.com/10.1007/978-3-319-01857-7

[25] W. K. Mennell, "HEURISTICS FOR SOLVING THREE ROUTING PROBLEMS : CLOSE-ENOUGH TRAVELING SALESMAN PROBLEM , CLOSE-ENOUGH VEHICLE ROUTING PROBLEM , SEQUENCE-DEPENDENT TEAM ORIENTEERING PROBLEM," PhD, University of Maryland, 2009. [Online]. Available: http://hdl.handle.net/1903/9822

[26] D. A. Grundel and D. E. Jeffcoat, "Formulation and solution of the target visitation problem," in *Proceedings of the AIAA 1st Intelligent Systems Technical Conference*, 2004.

[27] A. Arulselvan, C. W. Commander, and P. M. Pardalos, "A hybrid genetic algorithm for the target visitation problem," Tech. Rep., 2007.

[28] K. Sylejmani, J. Dorn, and N. Musliu, "A Tabu Search approach for Multi Constrained Team Orienteering Problem and its application in touristic trip planning," in *2012 12th International Conference on Hybrid Intelligent Systems (HIS)*. IEEE, Dec. 2012, pp. 300–305. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6421351

[29] X. Wang, B. L. Golden, and E. A. Wasil, "Using a genetic algorithm to solve the generalized orienteering problem," in *The Vehicle Routing Problem: Latest Advances and New Challenges*, B. L. Golden, E. Wasil, and S. Raghavan, Eds. Springer, 2008, pp. 263–273.

[30] Z. W. Geem, C.-l. Tseng, and Y. Park, "Harmony Search for Generalized Orienteering Problem : Best Touring in China," in *Advances in Natural Computation*, 2005, pp. 741–750.

[31] M. J. Broxton and L. J. Edwards, "The Ames Stereo Pipeline: Automated 3D Surface Reconstruction from Orbital Imagery," in *Lunar and Planetary Science Conference*, 2008.

[32] Z. M. Moratto, M. J. Broxton, R. A. Beyer, M. Lundy, and K. Husmann, "Ames Stereo Pipeline, NASA's Open Source Automated Stereogrammetry Software," in *Lunar and Planetary Science Conference*, 2010.

[33] S. W. Squyres, A. H. Knoll, R. E. Arvidson, J. W. Ashley, J. F. I. I. I. Bell, W. M. Calvin, P. R. Christensen, B. C. Clark, B. A. Cohen, P. A. de Souza, L. Edgar, W. H. Farrand, I. Fleischer, R. Gellert, M. P. Golombeck, J. Grant, J. Grotzinger, A. Hayes, H. Herkenhoff, J. R. Johnson, B. Jolliff, G. Klingelhofer, A. Knudson, R. Li, T. J. McCoy, S. M. McLennan, D. W. Ming, D. W. Mittlefehldt, R. V. Morris, J. W. J. Rice, C. Schroder, R. J. Sullivan, A. Yen, and R. A. Yingst, "Exploration of Victoria Crater by the Mars rover Opportunity," *Science*, vol. 324, no. 5930, pp. 1058–1061, 2009.

[34] R. Li, K. Di, J. W. Hwangbo, Y. Chen, and A. S. Team, "Rigorous Photogrammetric Processing of HiRISE Stereo Images and Topographic Mapping at Mars Exploration Rover Landing Sites," in *LPSC*, League City, TX, 2008.

[35] R. Li, S. W. Squyres, R. Arvidson, B. Archinal, J. Bell, Y. Cheng, L. Crumpler, D. J. Des Marais, K. Di, T. A. Ely, M. Golombek, E. Graat, J. Grant, J. Guinn, A. Johnson, R. Greeley, R. Kirk, M. Maimone, L. H. Matthies, M. Malin, T. Parker, M. Sims, L. A. Soderblom, S. Thompson, J. Wang, P. Whelley, and F. Xu, "Initial Results of Rover Localization and Topographic Mapping for the 2003 {Mars Exploration Rover} Mission," *Photogrammetric Engineering & Remote Sensing*, vol. 71, no. 10, pp. 1129–1142, 2005.

[36] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo Tourism: Exploring image collections in 3D," in *Proceedings of SIGGRAPH*, 2006.

[37] ——, "Modeling the World from Internet Photo Collections," *International Journal of Computer Vision*, 2007.

[38] Y. Furukawa and J. Ponce, "Accurate, Dense, and Robust Multi-View Stereopsis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1362–1376, 2010.

[39] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski, "Towards Internet-scale Multi-view Stereo," in *CVPR*, 2010.

[40] "King's bowl," https://www.google.com/earth/, 2014.

[41] King's Bowl, "42.949475 degrees N and -113.216297 degrees W," Google Earth, August 2013, accessed 12 January 2015.

[42] LROC Team, "LRO-L-LROC-2-EDR-V1.0," ARIZONA STATE UNIVERSITY, EDR M126759036L, 2013.

[43] C. H. Acton, "Ancillary data services of NASA's navigation and ancillary information facility," *Planetary and Space Science*, vol. 44, no. 1, pp. 65–70, 1996.

[44] T. Moller and B. Trumbore, "Fast, minimum storage ray/triangle intersection," *Journal of Graphics Tools*, 1997.

[45] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Single-Source Shortest Paths*, 2nd ed. MIT Press, 2001, ch. 24, pp. 580–619.

[46] S. Fuhrmann, F. Langguth, and M. Goesele, "Mve - a multi-view reconstruction environment," in *Proceedings of the Eurographics Workshop on Graphics and Cultural Heritage*, 2014.

[47] C. Wu, "Towards linear-time incremental structure from motion," in *3DV*, 2013.

[48] C. Wu, S. Agarwal, B. Curless, and S. M. Seits, "Multicore bundle adjustment," in *CVPR*, 2011.

[49] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski, "Towards internet-scale multi-view stereo," in *CVPR*, 2010.

[50] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multi-view stereopsis," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1362–1376, 2010.

[51] M. Lionel, P. Moulon, and M. Pascal, "Automatic

homographic registration of a pair of images with a contrario elimination of outliers," in *IPOL*, 2012.

[52] N. Snavely, S. Seitz, and R. Szeliski, "Photo tourism: Exploring image collections in 3d," in *SIGGRAPH*, 2006.

[53] ——, "Modeling the world from internet photo collections," 2007.

[54] D. Martinec, "Robust multiview reconstruction," 2008.

[55] A. Chatterjee and V. M. Govindu, "Efficient and robust large-scale rotation averaging," in *ICCV*, 2013.

[56] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," in *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1992.

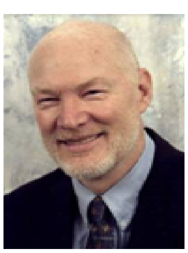
[57] C. Yang and G. Medioni, "Object modelling by registration of multiple range images," in *Image Vision Computing*, 1991.

[58] Z. Zhengyou, "Iterative point matching for registration of free-form curves and surfaces," in *International Journal of Computer Vision*, 1994.

BIOGRAPHY

Heather Jones *received her B.S. in Engineering and B.A. in Computer Science from Swarthmore College in 2006. She analyzed operations for the Canadian robotic arm on the International Space Station while working at the NASA Johnson Space Center. She is pursuing a PhD in Robotics at Carnegie Mellon University, where she researches reconnaissance, exploration and modeling of planetary pits, as well as planning and navigation for planetary robots.*

Wennie Tabib *received her B.S. degree in Computer Science at Carnegie Mellon University in 2012 and her M.S. degree in Robotics at Carnegie Mellon University in 2014. She developed software to optically register to terrain while working at Astrobotic Technology, Inc. She is currently pursuing a PhD in Computer Science at Carnegie Mellon.*



William L. "Red" Whittaker *received his B.S. from Princeton University, and his M.S. and PhD from Carnegie Mellon University. He is a University Professor and Director of the Field Robotics Center at Carnegie Mellon. Red is a member of the National Academy of Engineering, and a Fellow of the American Association for Artificial Intelligence.*