Scientist-Guided Autonomy for Self-Reliant Rovers

Gary Doran **Umaa Rebbapragada** **Eugenie Song** **Kiri Wagstaff**

Daniel Gaines **Robert Anderson** **Ashwin Vasavada**

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, California 91109
{firstname.lastname}@jpl.nasa.gov

Abstract

Although Mars rover missions have been highly successful in accomplishing scientific objectives, mission productivity is limited by challenges stemming from the need for commanding ground-based targeted observations under communication constraints imposed by the large distance between Earth and Mars. With an aging fleet of sun-synchronous relay orbiters, the opportunities for regular communication with rovers may become even more limited. In addition to onboard planning, robust navigation, and health assessment, there are strategies to make future rovers more self-reliant by enabling them to perform autonomous scientific characterizations of new areas during periods without an opportunity for ground-based targeted observations. In particular, we have studied how a "walkabout" strategy, in which an initial high-level characterization of a region is used to inform subsequent passes with specific targeted observations, was used successfully during the investigation of Pahrump Hills by the Mars Science Laboratory rover. Inspired by this approach, we have identified several capabilities that could allow a rover to autonomously perform some of these initial high-level characterization steps. In this paper, we describe technologies for identifying specific geologic units, regions, or features of interest, identifying areas of contact between two adjacent units, detecting and determining the orientation of layering within rock units, identifying novel and interesting features, and planning observations of regions with different sampling strategies using remote sensing instruments. The observations acquired with these approaches are driven by scientists' guidance and can provide scientists with data to help inform their decisions about where to make more resource-intensive targeted observations.

Introduction

Among the many challenges of operating a mobile spacecraft on the surface of another planet are constraints on communication with the vehicle. Due to the inevitable delay in communications traveling over interplanetary distances at the speed of light, direct remote operation is not possible. Instead, for the Mars Science Laboratory (MSL) *Curiosity* rover, scientists and operators typically pre-plan at least one *sol*, or Martian day, of activities for the rover to perform. This plan is uplinked to the rover, which carries out activities and transmits science results, telemetry, and images back to Earth through relay orbiters such as the Mars Reconnaissance Orbiter (MRO). Operators usually do not begin planning for the next sol's activities until they have received updated rover state information, especially after the rover has driven to a new location. Current relay orbiters like MRO provide regular passes several times every day. However, other relay orbiters that would act as a replacement for MRO in case of any failure do not have orbits with regular over-flights. With these replacement relay orbiters, it could be several sols before the rover is able to communicate its status back to Earth.

To ease the challenges caused by the need for regular communications to operate a rover, we are exploring technologies to make future rovers more "self-reliant." Self-reliant rovers should be able to take a set of high-level goals rather than specific, precisely defined activities, and reliably carry out the operators' intent even after several sols without communication with Earth. Technologies to enable these goals include robust navigation, onboard health assessment, localization, and onboard planning. In this paper, we focus on a set of technologies that enable satisfying high-level science objectives. First, we describe the results of a case study in which we identified the need for specific capabilities to achieve high-level science objectives when ground-targeted activities are not possible (Gaines et al. 2016). Then, we discuss several existing technologies currently used for science autonomy onboard MSL and other missions. Finally, we discuss new technologies we have developed to enable accomplishing science objectives.

Case Study: Pahrump Hills

On Sol 780, MSL arrived at an area called Pahrump Hills containing exposed outcrops of sedimentary rock (Stack et al. 2015). Scientist knew that they wanted to spend some time taking detailed measurements such as drill samples at several locations at Pahrump Hills, but more information would be useful in determining the best places to acquire these measurements. The scientists devised a "walkabout" strategy in which they would drive around the Pahrump Hills area several times. The first pass would be devoted to performing a high-level characterization of the area with high resolution imagery of select, interesting portions of the outcrop and remote sensing observations using the ChemCam

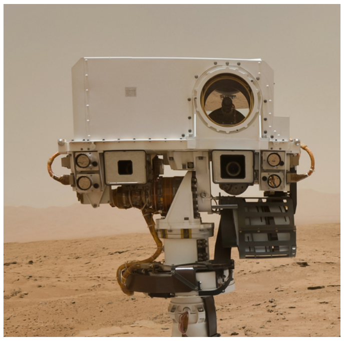

Figure 1: The remote sensing mast of the Mars Science Laboratory Curiosity rover contains the ChemCam and Mastcam instruments.

Laser-Induced Breakdown Spectroscopy (LIBS) instrument (Wiens et al. 2012) and the high resolution Mastcam cameras (Malin et al. 2005). Both instruments are located on the rover remote sensing mast (see Figure 1). ChemCam uses a laser to vaporize rock targets from a distance of several meters away and records the light emitted from the resulting plasma to determine the rock's elemental composition. Subsequent walkabout passes would be devoted to more time-consuming measurements and sampling activities of the most interesting locations identified during the first pass.

The walkabout strategy naturally lends itself to support by autonomous science capabilities. In particular, a self-reliant rover could spend several sols performing an initial area characterization as in the first pass of a walkabout when ground-based targeting is not possible. Armed with the information acquired during this first pass, scientists can command a set of specific measurements after the rover has had a chance to communicate its results to the Earth. To validate this strategy in a real-world operational scenario, autonomous characterization capabilities were recently employed successfully as part of a walkabout strategy during the CanMars 2016 Mars Sample Return Analogue Deployment in Utah (Francis, Gaines, and Osinski 2017).

Some capabilities required for initial area characterization during the first pass of a walkabout include identifying regions of interest and planning observations of these regions using particular sampling strategies. Both what makes a region interesting and what strategy to use for sampling each region are driven by science objectives. Examples of interesting regions include specific geologic "units," "contacts" between two adjacent units (where they meet each other), geologic features such as veins of material within units, dust-free regions of a rock surface, or unexpected, novel features. Sampling strategies include selecting spatially diverse locations across a large span of exposed outcrop, taking LIBS measurements *across* layers of sedimentary rock to see how composition varies between layers, or following a vein or contact along its length with several LIBS measurements or images. We describe technologies to enable these capabilities below.

Related Work

The new technologies we propose to enable self-reliant rovers can be seen as an extension of existing, successful approaches to scientist-guided autonomy for increasing the productivity of surface and orbital exploration missions. The Autonomous Exploration for Gathering Increased Science (AEGIS) system, which was developed for the Mars Exploration Rover (MER) *Opportunity* (Estlin et al. 2012) and was recently deployed for operational use on on MSL (Francis et al. 2016), is one key example. AEGIS uses a specialized algorithm called Rock Segmentation Through Edge Regrouping (Rockster) to locate rocks, patches of outcrop, or veins within an image (Burl et al. 2016). Then, ranking rules use features such as size, shape, and albedo to select targets according to scientists' preferences. AEGIS typically runs on board after a drive, but before operators on the ground have had a chance to select specific targets for measurement in the new environment. On MSL, AEGIS triggers follow-up measurements with the ChemCam instrument, providing scientists with an additional level of characterization of a new area (beyond standard post-drive imaging) before any targeted observations are possible.

Another key technology used to enable autonomous exploration is the TextureCam software (Thompson et al. 2012). TextureCam is a pixel-wise classification algorithm that uses a random forest model trained on a set of labeled images. The generality of TextureCam allows it to be used for identifying everything from specific geologic units for surface missions to atmospheric features such as clouds seen from space during orbital missions (Chien et al. 2016). Many of the new technologies we propose use the output of TextureCam, a probability that each pixel belongs to a class of interest, as an input for processing and target selection.

Finally, while TextureCam can be used to identify known regions of interest for which labeled training images exist, there are other geologic features that might be interesting for their *novelty*. To identify such features, we explore the use of several novelty detection algorithms. One such algorithm is Discovery through Eigenbasis Modeling of Uninteresting Data (DEMUD), which has been successfully used for novelty detection in ChemCam LIBS spectra (Wagstaff, Lanza, and Wiens 2014). DEMUD works by building a model of "uninteresting" data; that is, data that has already been observed and characterized. Any new data that is unlikely under this model is flagged as "interesting," then incorporated into the model of uninteresting data. By continually adjusting the model, DEMUD learns to expect examples after they have been observed and are thus no longer novel. Another algorithm explored for novelty detection is the Isolation Forest (Liu, Ting, and Zhou 2008). An isolation forest uses the average number of random splits in a set of decision trees required to separate one example from the others as a measure of that example's novelty. The approach assumes that novel examples can be separated from the rest of the data with relatively few splits. We explore both DEMUD and isolation forests as alternative approaches to novelty detection.

Novel Tools For Scientist-Guided Autonomy

Now, we describe building upon existing work to develop new technologies for characterizing the geology of a new area on Mars. Generally, these technologies fall into two categories: those that enable finding specific targets or regions of interest and those that enable designing follow-up measurements given selected targets. Navigation camera or Navcam images are used for finding regions of interest, and Navcam panoramas are typically acquired after driving into any new area. We assume that such images are available as an input to autonomous science processing. We begin by describing a technology for finding a specific type of interesting region, a "contact" between two adjacent geologic units.

FORC: Finding Oriented Regions of Contact

A geologic contact between two units can reveal key information about the history of formation of the units as well as what occurred in the intervening time between when each of the units was deposited. In principle, software such as TextureCam can be used to directly identify a general region where a contact between two units occurs. However, challenges prohibit this approach in practice. In particular, TextureCam requires training data consisting of example images of the region of interest. However, it is possible that while many examples of two units have been seen, there has not yet been a clear example of the contact between the two units. A more robust approach might first identify geologic units and then search for contacts between them.

We have developed a new algorithm called Finding Oriented Regions of Contact (FORC) that implements the approach of identifying contacts after identifying individual units. For geologic units A and B, FORC works by first running TextureCam on board to produce probability maps $\mathbf{P}_A$ and $\mathbf{P}_B$ containing the probabilities that each pixel in an image $\mathbf{I}$ belongs to units A or B, respectively. We assume that the orientation of the contact between A and B (that is, whether A is above or below B stratigraphically) is known *a priori* from orbital context images. The FORC algorithm allows scientists to specify an arbitrary orientation angle θ between the two contacts. For example $\theta = 90^\circ$ corresponds to B being directly above A. Scientists also specify a scale in pixels of the size of the minimum size of the rock units expected to be exposed around the contact. Then, $\mathbf{P}_A$ and $\mathbf{P}_B$ are translated by this amount in the direction θ and $-\theta$, respectively. The resulting translated matrices are multiplied component-wise. This corresponds to the probability that each pixel is in the direction of θ from unit A and in the direction of $-\theta$ from unit B. Pixels with a high value of this probability are located within the oriented contact region. This probability is called the "contact score." After the contact score is derived, any number of algorithms (some of which are described below) can be used to select targets for observation based on this score.

An example using the FORC algorithm is shown in Figure 2. At an area known as Marias Pass, the Curiosity rover encountered a contact between the ligher-toned Murray unit and darker-toned Stimson unit (Newsom et al. 2016). Using previously acquired Navcam images, we trained TextureCam to label each of these units. Then, we applied the trained models on an image of the contact from Sol 992. Using the output of TextureCam, the FORC algorithm was then used to produce a contact score, shown on the far right of Figure 2. The pixels with contact score exceeding 0.5 are highlighted. Although there are some small false-positive regions, the approach successfully identifies much of the visible contact region between the two units.

Finding the precise location of a contact can be challenging, especially using low-resolution Navcam images. However, FORC can be used to command follow-up Mastcam images of a region where there is a change between two units or rock textures. Either the contact is likely to be contained within this region, which scientists can examine on the ground, or the region contains a "facies change" representing a significant change in the process that formed the rock. Both types of regions are scientifically interesting and can be better understood by scientists using high-resolution follow-up images.

FOLD: Fast Oriented-Layer Detector

Layering or stratification is another essential feature used to geologically characterize a new area. Layered rocks can be formed by the deposition of materials via water, air, or other geological processes such as lava flows. Understanding the depositional environment of a rock can inform the conditions present at the time of its formation, which can give clues about past habitability.

Existing approaches such as TextureCam can be used to directly identify regions with layering (Wagstaff et al. 2013). However, in addition to identifying these regions, it is also desirable to infer the *orientation* of layering within each region to inform follow-up measurement strategies. For example, it could be desirable to acquire LIBS measurements *across* the layers within a rock to determine how the depositional environment changed over time. TextureCam is not directly capable of determining layer orientation because it operates at the single-pixel level, so it cannot inform how to acquire a measurement of an identified target.

The Fast Oriented-Layer Detector (FOLD) algorithm is designed to both detect and determine the orientation of layers within a scene. FOLD works by first computing the gradient of an image, which quantifies the magnitude and direction of change in image pixel intensities at each location. Then, within a particular region of the image, a histogram of gradient directions weighted by gradient magnitude is computed. A probabilistic mixture model, composed of a Gaussian "signal" component and a uniform "noise" component, is then fit to the data. The resulting maximum-likelihood (ML) model produces an effective signal-to-noise ratio (SNR) for the detection, as well as an estimate of the layer orientation. By using a histogram-based approximation of the distribution of gradient directions, the algorithm's time and memory complexity is linear with respect to the size of the input image.

For the first step of gradient computation, it is beneficial to first smooth a 16-bit version of the image using a Gaussian kernel. To reduce memory consumption, the gradient magnitude is represented using a 16-bit integer value, scaled down by a factor of $\sqrt{2}$. Similarly, the gradient angle is represented using an 8-bit integer, representing one of 256 evenly spaced "bins" between 0 and π. Gradient directions are considered equivalent up to the sign of the gradient vector.

After the gradient is computed for each pixel in the image, a histogram is computed for some subset of the pixels in the image corresponding to a region of interest. Since the

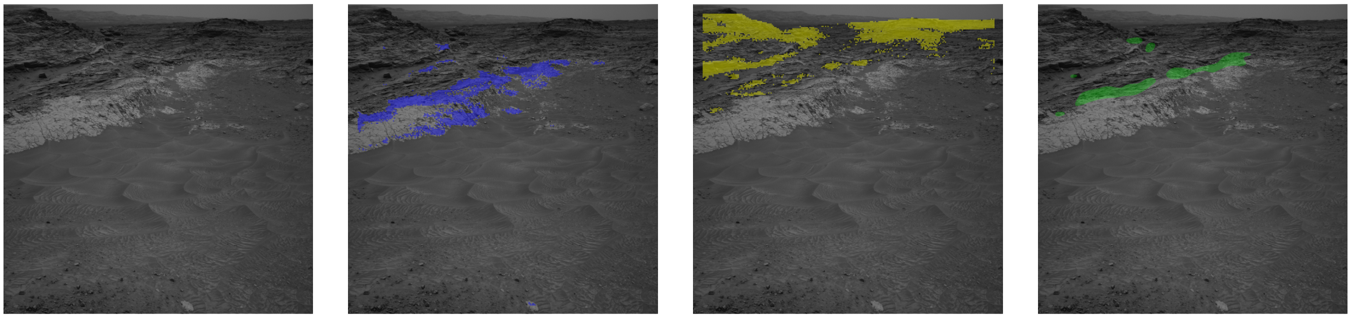

Figure 2: **Left:** The Murray–Stimson contact at Marias Pass. **Center:** The Murray (blue) and Stimson (yellow) units are identified using TextureCam. **Right:** FORC is used to derive a contact score, with the highest-valued regions highlighted (green).

gradient angles have already been discretized into 256 bins, these are the bins used for the histogram. The gradient magnitude is used as a weight for each pixel's corresponding gradient direction, because the largest gradients correspond to the strongest visual evidence of layering.

Given a histogram for a region of interest, the goal is to fit a probabilistic model that treats the gradient angle distribution as a mixture of a uniform noise distribution and a Gaussian distribution centered about the direction of layering. Formally, this model is

$$\mathrm{P}\left(\theta \mid \alpha, \mu, \sigma\right) = \alpha\,\mathcal{N}_{[0,\pi]}\left(\theta \mid \mu, \sigma\right) + (1-\alpha)\,\mathcal{U}_{[0,\pi]}\left(\theta\right),$$

where α is the fraction of pixels coming from the layering distribution, $\mathcal{N}_{[0,\pi]}$ is the "wrapped" normal distribution with parameters μ and σ, and $\mathcal{U}_{[0,\pi]}$ is the uniform distribution over $[0, \pi]$. Because angles only take values on the interval $[0, \pi]$, the wrapped normal distribution, defined as

$$\mathcal{N}_{[0,\pi]}\left(\theta \mid \mu, \sigma\right) = \frac{1}{\sqrt{2\pi}\sigma} \sum_{k=-\infty}^{\infty} e^{-((\theta+k\pi)-\mu)^2/2\sigma^2},$$

is used to "wrap" the normal distribution within that interval. In practice, the terms of the infinite sum become negligible when $|k|$ exceeds a small multiple of σ/π.

An expectation–maximization (EM) approach is used to determine a maximum-likelihood (ML) estimate of the parameters α, μ, and σ. Given a histogram bin representing angle θ_i with probability mass m_i, we start by fixing the distribution parameters and computing weights w_i, which correspond to the estimate of the fraction of the pixels in that bin coming from the layering distribution $\mathcal{N}_{[0,\pi]}$. This is the *E-step*:

$$w_i \leftarrow \frac{\mathcal{N}_{[0,\pi]}\left(\theta_i \mid \mu, \sigma\right)}{\mathcal{N}_{[0,\pi]}\left(\theta_i \mid \mu, \sigma\right) + \mathcal{U}_{[0,\pi]}\left(\theta_i\right)}.$$

For the first iteration, these weights w_i are all set to 0.5.

Then, fixing the weights, the ML estimates of the other parameters are computed. This is the *M-step*:

$$\alpha \leftarrow \sum_i w_i m_i$$

$$\overline{x} \leftarrow \frac{\sum_i w_i m_i \cos(2\theta_i)}{\sum_i w_i m_i} \qquad \overline{y} \leftarrow \frac{\sum_i w_i m_i \sin(2\theta_i)}{\sum_i w_i m_i}$$

$$\mu \leftarrow \tfrac{1}{2}\operatorname{arctan}_2\left(\overline{y}, \overline{x}\right) \qquad \sigma \leftarrow \sqrt{-\tfrac{1}{2}\ln\left(\overline{x}^2 + \overline{y}^2\right)}.$$

These two steps repeat until the distribution parameters converge, or a maximum number of iterations has been exceeded. In practice, only on the order of 10 iterations are required for convergence. After convergence, an "SNR" estimate for the detection can be computed as $\frac{\alpha}{1-\alpha}$.

There are two primary modes of operation for FOLD. First, the above procedure can be run for a single rectangular region within the image, generating a single estimate of the SNR and average orientation μ within that region. This is useful for small field-of-view instruments that might have a layered rock fill the area they observe. However, for wide field-of-view instruments, a second mode of operation allows sliding a small rectangular window across the image and repeating the analysis for each pixel (or every n^{th} pixel) within the image. This allows generating a heat map of regions within the image likely to contain layered targets and a map of the layer orientation at each point within the image.

Figure 3 shows the output of FOLD operating in "single-region" mode for a small field-of-view ChemCam Remote Micro-Imager (RMI) image. On the left-hand side, the original image is shown with a solid line illustrating the orientation of the inferred lamination direction μ, along with dashed lines illustrating $\mu \pm \sigma$. Since the RMI has a circular field of view, a circular region is used when computing the histogram. On the right, the 256-bin histogram is shown along with the derived ML model in blue. The red lines correspond to the values shown in the left-hand plot. The SNR, derived from the α parameter, is shown at the top of the plot.

To evaluate the discriminative power of SNR for determining when layering is present in an RMI image, a sample of 50 RMI images were selected for evaluation. Of these, 25 contain visible layering while 25 do not. Using only a threshold on SNR to distinguish layered from unlayered rocks, it is possible to achieve an area under receiver operating characteristic (ROC) curve (AUC) of 0.922.

Figure 4 shows an example of FOLD operating in "sliding-window" mode. The image was taken by the MSL navigation camera, and shows a layered butte within the "Murray Buttes" region of Gale Crater. The lower image in the figure shows a heat map (white is the "hottest") of the SNR at each pixel within the image. Small black bars show the orientation of layering at each pixel. FOLD not only detects the layering within the butte, but also the layering of

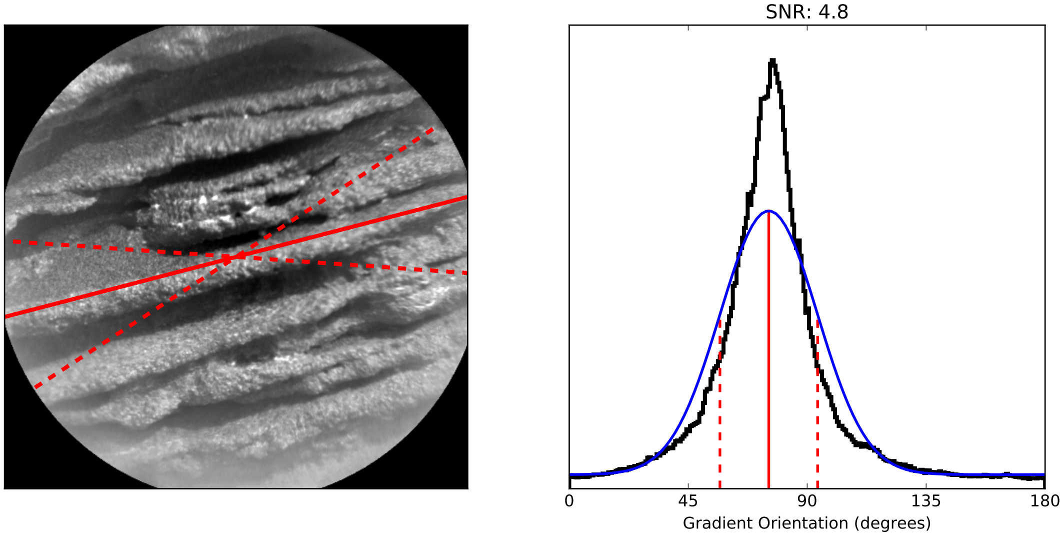

Figure 3: **Left:** The layer orientation inferred by FOLD (solid line) with one standard deviation (dashed line) shown over a small field-of-view RMI image. **Right:** The empirical histogram of gradients (black) and ML model (blue).

blocks that have fallen from their original locations.

Novelty Detection

In addition to identifying specific geologic units and features known to be interesting, there is also a need to identify novel characteristics of an area that have not been seen in previously-explored areas or stand out from what is expected. For example, features such as concretions, veins, or meteorites are worthy of remote sensing measurements during an initial characterization.

We have explored two algorithms for novelty detection: DEMUD and isolation forests. These algorithms can be applied in a straightforward way to Navcam images using a sliding window approach. Both algorithms can be trained on a set of previously-observed images to learn what is no longer novel. Then, applying a learned model produces an outlier score for each location as a sliding window is moved over a new image on board.

Example results for the two algorithms are shown in Figure 5. While future experiments will be aimed at characterizing the performance of these approaches under real-world operational conditions, preliminary results show that outlier detection approaches do select visually salient regions in images that are unlike the background sand or gravel. The isolation forest performs well when the raw pixel intensities within the window are used as feature vectors, whereas DEMUD performs better when higher-level histogram of oriented gradients (HOG) features are extracted from the window. Since HOG features could be more expensive to compute, isolation forests likely have more favorable computational performance for onboard novelty detection.

DOTS: Diverse Onboard Target Selection

After TextureCam, FORC, FOLD, or another algorithm selects a region of interest for measurement, a self-reliant rover must select specific points for measurement with instruments like ChemCam. The Diverse Onboard Target Selection (DOTS) algorithm addresses the scenario in which scientists are interested in acquiring "diverse" measurements of a geologic region. That is, if TextureCam detects several similarly relevant but disjoint geologic regions within a scene, scientists might prefer to take a measurement of each separate region rather than taking several measurements within one region. Such a measurement strategy allows scientists to assess the geochemical diversity of a scene rather than focusing measurements on one isolated area.

DOTS is designed to take pixel-wise scores such as TextureCam probabilities or FORC contact scores and produce a set of specific point targets to be measured. DOTS works by greedily selecting a target that has the highest probability of belonging to a region of interest, then removing from consideration all points that can be reached from the target by a path that stays (with high probability) inside a contiguous patch of the same region. This process continues until the desired number of targets have been found, or no relevant geological regions remain unmeasured.

As an example, suppose scientists are interested in autonomously measuring gray, freshly exposed rock that is broken after the rover drives over it (see Figure 6). TextureCam can be trained on the ground using previous example images of broken rocks. The trained model can be uplinked to the rover to predict the probability that each pixel in a new image corresponds to the same feature. Given the probability map $\mathbf{P}_{ij}$, DOTS proceeds by first processing this probability map to find sufficiently large regions of interest. This is accomplished by using a Gaussian filter with width σ corresponding to the minimum desired region size. This smoothing process ensures that any larger regions with consistently high probability of belonging to the class of interest will be preferred over smaller regions. Let the smoothed map pro-

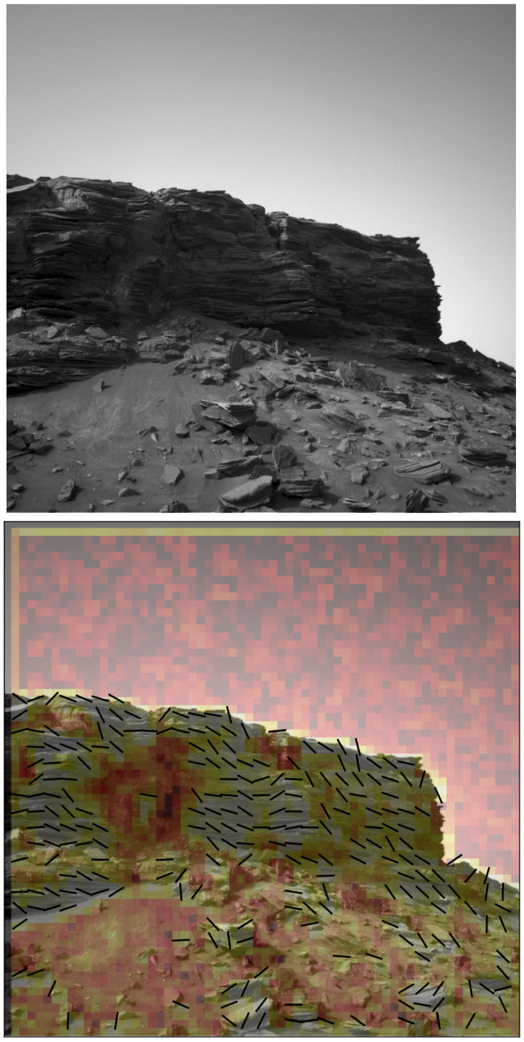

Figure 4: **Top:** MSL Navcam image of a butte in the Murray Buttes region of Gale Crater. **Bottom:** FOLD results, with hotter colors indicating higher SNR for layer detection and black bars showing the inferred orientation for the regions of the image with the highest SNR.

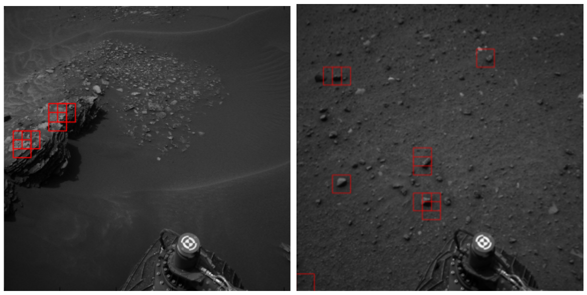

Figure 5: Example novelty detection results using the DEMUD algorithm (**Left**) and isolation forests (**Right**). Red boxes indicate regions where a sliding window had a high outlier score.

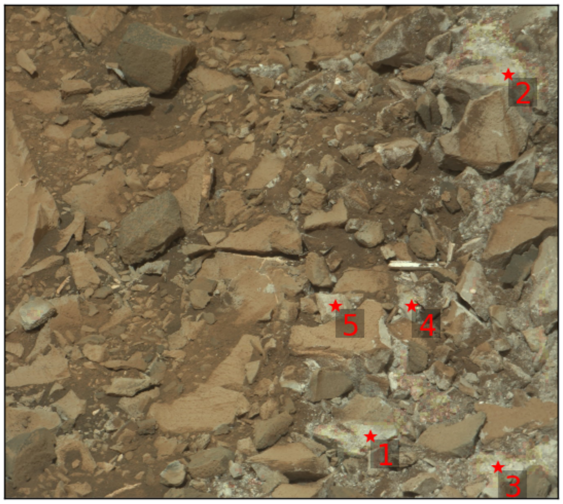

Figure 6: Example image showing targets selected by DOTS to measure freshly exposed gray rock. Five proposed targets are indicated with red marks, order from the highest priority (1) to the lowest priority (5).

duced from $\mathbf{P}_{ij}$ be called $\mathbf{S}_{ij}$.

Next, DOTS selects the point in the smoothed map $\mathbf{S}_{ij}$ with the largest value. This point (u, v) is chosen as the first target, unless the value falls below some specified threshold (and is therefore unlikely to belong to the region of interest).

If the point is selected as a target, then a linkage mask $\mathbf{L}_{ij}$ is computed for all pixels in the image. The linkage mask at (i, j) is defined as the minimum value of $\mathbf{S}$ along a path from (i, j) to (u, v), where the path is selected to maximize this value. The mask $\mathbf{L}_{ij}$ can be computed using a modified version of Dijkstra's algorithm. First, the point (u, v) is added to a max-on-top heap structure with value $\mathbf{S}_{uv}$. At each iteration, the highest-valued unvisited point is popped from the heap, and each of its neighbors are added to the heap with a value equal to the minimum of the heap value and the neighbor's value. This continues until all pixels have been visited, or the value at the top of the heap is sufficiently close to zero.

The linkage mask represents a lower bound on the probability that any pixel in the image is connected to the target pixel (u, v) by a path that remains with the region of interest. In the next iteration of target selection, it is desired to exclude such pixels. Therefore, the smoothed map $\mathbf{S}$ is updated using the rule $\mathbf{S} \leftarrow \mathbf{S} \cdot (1 - \mathbf{L})$ since $1 - \mathbf{L}$ corresponds to the logical `NOT` operation applied to the linkage map. That is, this update rule produces a map of regions of interest that are not linked to the previous target.

After the update, the process is repeated until the desired number of targets are selected, or all pixels in $\mathbf{S}$ are below some threshold (i.e., there are no more relevant targets to select). As shown in Figure 6, DOTS does in fact select diverse pieces of freshly exposed rock rather than targeting multiple measurements on the same piece of broken rock.

OnRAMP: Onboard Raster and Mosaic Planner

With the ChemCam LIBS instrument, it is common to take a series of measurements in a line or grid to cover a region of interest. Similarly a Mastcam mosaic comprised of several images can be used to document a region that is too large for a single image frame. In both cases, enabling a rover to emulate ground-targeted behavior requires the capability to plan rasters and mosaics on board. We have developed the Onboard Raster And Mosaic Planner (OnRAMP) algorithm to provide this capability. As with DOTS, OnRAMP works by using the output of other software such as TextureCam to identify a feature or region of interest. Then, using a dynamic programming algorithm, OnRAMP plans a series of point measurements or image frames to construct a raster or mosaic that optimally measures the identified feature or region. OnRAMP accounts for instrument pointing uncertainty and constraints as part of its optimization. OnRAMP is designed to run efficiently onboard a spacecraft with limited memory and computational requirements.

OnRAMP operates in two modes: planning a raster and planning a mosaic. However, the fundamental approach is the same for each mode. We begin by describing the process of planning a raster. As an example, scientists might be interested in identifying veins to measure within the field-of-view of the ChemCam RMI (see Figure 7, left). Using previously acquired images of similar features, scientists can train a TextureCam model to identify veins within new images. More precisely, TextureCam produces a map of the probability that each pixel in the image corresponds to a vein. An example is shown in Figure 7 (center).

Given the probability map produced by TextureCam, the next step is to translate the probability of selecting a particular location for measurement to the probability that the feature of interest will actually be measured, taking instrument pointing error into account. Assuming Gaussian pointing error, this can be accomplished by convolving a Gaussian filter with the probability map. The filter should have a width corresponding to a 1-σ pointing error. An example is shown in Figure 7 (right).

After the filtered probability map is computed, the individual raster points can be planned. The raster points are selected such that the instrument only slews monotonically along each of the pointing axes (azimuth and elevation). For ChemCam, this direction is typically up and to the left (negative azimuth and positive elevation). Slewing in one direction leads to more accurate pointing because it avoids "backlash" in the gearing mechanisms that drive the pointing of the instrument mast. Furthermore, it is desirable to require some minimum azimuth and elevation separation between targets, in case the instrument slightly overshoots a target due to pointing error. For example, the minimum separation can be set equal to the 3σ pointing error so that even if one target is missed, the instrument can continue slewing in the same direction to reach the next target.

The raster points can be selected using a recursive algorithm made efficient with dynamic programming. The recursive nature of the solution follows from the following observations. First, for a 1-point raster, the expected value of selecting a particular point in the image to measure is simply the value of that point in the filtered probability map (as in Figure 7, right). Then, for an optimal k-point raster, the value of starting at any point in the image is equal to the value of that point in the probability map plus the maximum value of any $(k-1)$-point raster that starts up and to the left of that k^{th} point by the minimum separation between points.

The simple recurrence relationship described above can be used to design a dynamic programming algorithm to find an optimal set of raster points. Let $\mathbf{P}$ be the filtered probability, with $\mathbf{P}_{ij}$ the value at row i, column j. Let $\mathbf{R}^k_{ij}$ be the value of starting a k-point raster at point (i,j). The base case is a 1-point raster, for which $\mathbf{R}^1 = \mathbf{P}$. Then, if the minimum point separation is s pixels, the recurrence relation is:

$$\mathbf{R}^k_{ij} \leftarrow \mathbf{P}_{ij} + \max_{\substack{0 \le u \le i-s,\\ 0 \le v \le j-s}} \mathbf{R}^{k-1}_{uv}. \tag{1}$$

We can use a memoization process to compute this recurrence efficiently. Starting with $\mathbf{R}^1$, we can use this relation to compute $\mathbf{R}^2$, followed by $\mathbf{R}^3$ and continuing until the desired $\mathbf{R}^n$ is computed for a n-point raster.

An example showing the memoization process is given in Figure 8. The process starts in the top left corner with the $\mathbf{R}^1 = \mathbf{P}$ matrix representing the value of choosing any single point for a raster. Moving left, $\mathbf{R}^2$ is computed using the expression in Figure 1. This continues until the lower right-most $\mathbf{R}^5$ raster value matrix is computed.

Selecting the raster points requires a final, second pass across the value matrices $\mathbf{R}^k$. First, the point p_1 with maximum value within $\mathbf{R}^n$ is chosen. Then, the point with maximum value in $\mathbf{R}^{n-1}$ that is up and left of p_1 by at least s pixels is selected as the second point, p_2. The process continues until all n points are selected. The running example with 5 points is shown in Figure 9.

Algorithm 1 OnRAMP algorithm for rasters

```
Require: Input image $\mathbf{I}$, minimum separation $s$, pointing
  error $\sigma$, raster points $n$

  $\mathbf{P} \leftarrow \mathrm{TextureCam}(\mathbf{I})$
  $\mathbf{P} \leftarrow \mathrm{GaussianFilter}(\mathbf{P}, \sigma)$
  $\mathbf{R}^1 \leftarrow \mathbf{P}$
  for $k \leftarrow \{2 \ldots n\}$ do
    $\mathbf{R}^k_{ij} \leftarrow \mathbf{P}_{ij} + \max_{\substack{0 \le u \le i-s,\\ 0 \le v \le j-s}} \mathbf{R}^{k-1}_{uv}$
  end for

  $p_{\text{last}} \leftarrow (\mathrm{Height}(\mathbf{I}), \mathrm{Width}(\mathbf{I}))$
  for $k \leftarrow \{n \ldots 1\}$ do
    $p_{n-k+1} \leftarrow \arg\max \left\{\mathbf{R}^k_{ij} \mid (i+s, j+s) \le p_{\text{last}}\right\}$
    $p_{\text{last}} \leftarrow p_{n-k+1}$
  end for
  return $\{p_1, \ldots, p_n\}$
```

An overview of the OnRAMP algorithm described above for planning rasters is shown in Algorithm 1. The algorithm takes an image, a minimum separation, pointing error σ, and

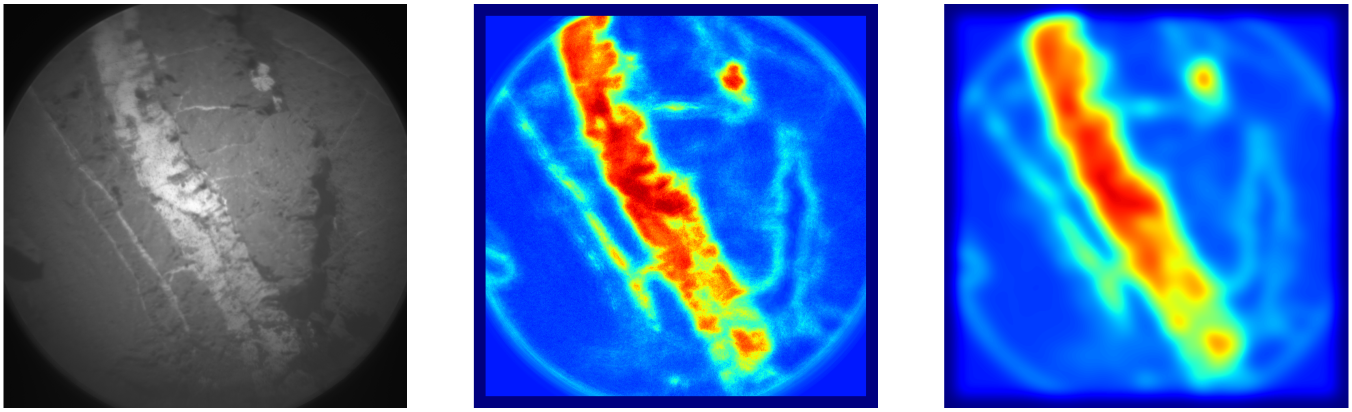

Figure 7: **Left:** A ChemCam RMI showing a small vein target, roughly one centimeter across. **Center:** A map showing the TextureCam-determined probability that each pixel corresponds to vein in the image. Red indicates higher values and blue indicates lower values. **Right:** Probability blurred with a Gaussian filter to account for instrument pointing error.

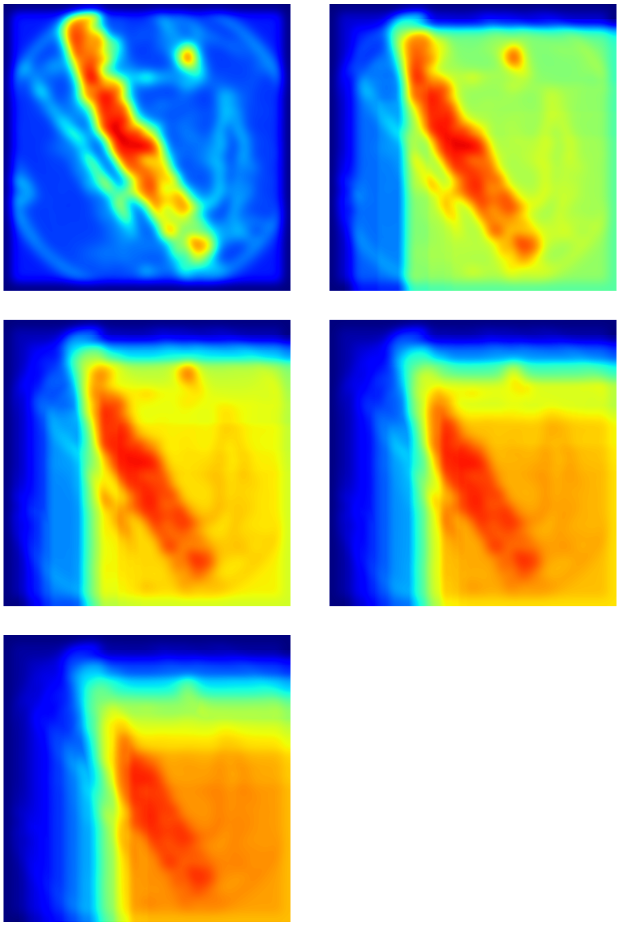

Figure 8: An example showing memoization to compute the value of choosing a point to begin a 1-point ($\mathbf{R}^1$, top left) to 5-point ($\mathbf{R}^5$, bottom right) raster.

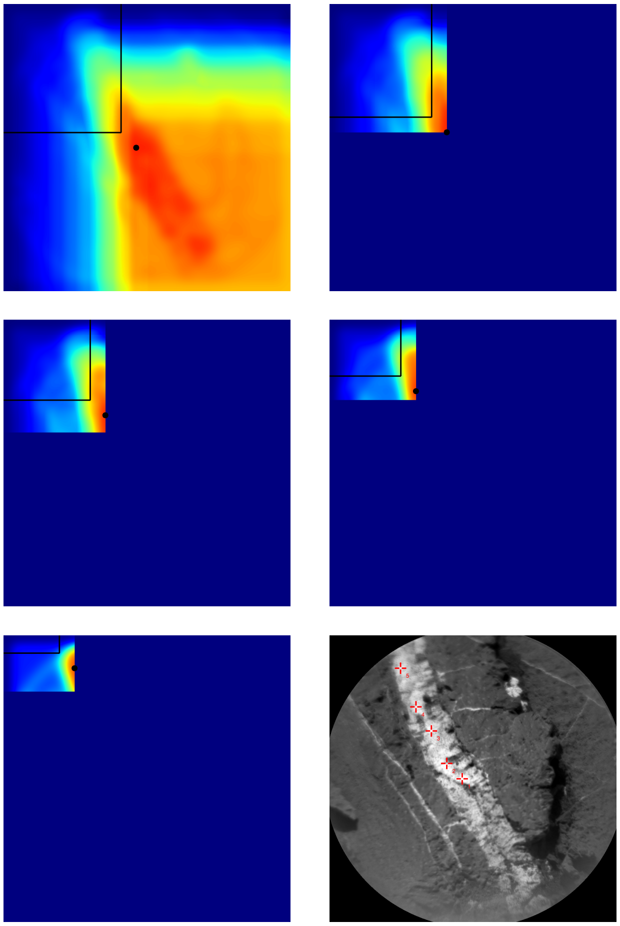

Figure 9: An example showing how raster points are selected, working backwards from the memoized value matrices. At each step, the black dot is selected as the k^{th} raster point, and the subsequent $k-1$ points must be in the region to the upper-left labeled with the black box. The resulting 5-point raster is shown in the lower right.

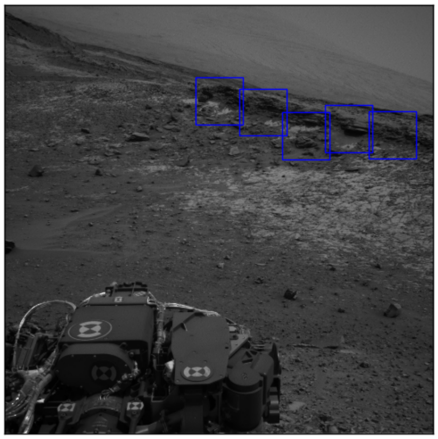

Figure 10: An example mosaic using the OnRAMP algorithm given a region of interest identified by finding the contact between two rock units.

desired raster points n, and it produces a set of raster points $\{p_1, \ldots, p_n\}$.

A few small adjustments to the algorithm above can be used to plan image mosaics instead of point-measurement rasters. First, the input image for a mosaic is likely to be a wide field-of-view Navcam image rather than an RMI. TextureCam or FORC can again be used to identify the likelihood that each pixel is a region of interest within the image.

Since an image covers a wide spatial area, rather than using a Gaussian filter, a Uniform filter is convolved with the TextureCam output and used to compute the expected percent of an image frame that contains the region of interest. The resulting map is the expected fraction of an image centered at each point that contains the region of interest. Thus, each point in the filtered image contains information about the value of all possible image frames of a given size. This process assumes that the Navcam and Mastcam are roughly co-boresighted so that the Mastcam frame is a rectangular region within the Navcam frame.

Selecting a sequence of Mastcam frames to cover a region is similar to designing a raster. The key difference is the constraint between selected frames. Instead of requiring separation by some minimum amount, a mosaic requires overlap by some minimum amount. So the value of a k-frame mosaic is the value of the k^{th} frame at a given point plus the value of the of a frame that is to the left of the k^{th} frame by a little less than a frame width, and translated vertically by some maximum amount. An example image sequence generated with this recurrence relation is shown in Figure 10. Although horizontal mosaics are more common, a different recurrence relation can be used to find vertical mosaics.

Use Cases

The set of tools described above can be used to enable scientist-guided autonomous exploration in new areas during periods when ground-based targeting is not possible due to communication constraints. Scientists can express their intent for observations using various combinations of these tools. A few examples illustrating how these tools can be used together are described below.

Diverse LIBS of a Geologic Unit

Suppose the rover is driving into an area with an exposed outcrop of bedrock that has been seen previously. Scientists might be interested in characterizing the composition of this bedrock for comparison to the composition in the previously observed area. Using previous image of this bedrock unit, scientists can train a TextureCam model to recognize the same unit in the new area. After driving into a new area, the standard post-drive images are analyzed to identify regions containing the bedrock, then DOTS is used to select several targets for measurement.

High Resolution Images of a Contact Region

Expanding upon the previous scenario, suppose that a new area contains two geologic units A and B that are in contact with each other, as seen from orbital imagery. Scientists can train TextureCam to recognize both A and B. Then, scientists can specify using the FORC algorithm that they are interested in regions where unit B overlies unit A. After finding such regions, scientists can specify that OnRAMP should be used to plan and acquire an image mosaic of the contact region.

LIBS of a Layered Unit

Suppose an area contains a known, layered unit, and scientists would like to acquire LIBS measurements *across* the layering to characterize changes in the depositional environment over time. Again, scientists can use either TextureCam or the FOLD algorithm to identify the unit of interest, then DOTS to select several targets for follow-up measurement with a LIBS instrument. For each LIBS measurement, the RMI camera can be used to determine the orientation of the layering, then a linear raster can be planned with points spaced in azimuth and elevation to traverse the layers.

Novelty Detection During a Strategic Drive

The rover might spend several sols driving from one area of interest to another during periods without ground-targeted observations. In such scenarios, it is important to acquire measurements of any novel or unexpected features, since it is unlikely for the rover to return for follow-up measurements. Scientists can specify their intent to search for novelty periodically during a drive. Then, either DEMUD or an isolation forest can be used to locate novel features during a drive and acquire either high-resolution images or LIBS measurements.

Future Work

We plan to test these capabilities on a testbed rover in the "Mars Yard" at JPL over the next year. The first test will involve a basic scenario to evaluate the FORC algorithm on a simulated contact region constructed using two different toned flagstone rocks. The contact will be identified, and several follow-up images with a mast camera will be acquired. After further development, the autonomous science tools will be integrated with the onboard planner to allow trading off between several science goals during characterization. Finally, a full integration with other self-reliant rover systems such as robust navigation will allow simulating scenarios in which scientist-guided autonomy is employed both during a strategic drive as well as after reaching a scientific area of interest. These demonstrations will enable an evaluation of the various scientist-guided autonomy tools in a realistic environment.

Conclusion

In addition to robust navigation and onboard goal planning, the ability to autonomously characterize the geology of an area of interest is important to increasing productivity of rover missions when there are limited opportunities for ground-in-the-loop commanding. We have developed a set of tools that can be used to enable such a characterization using guidance from scientists, who specify the types of features they are interested in observing, how to design measurements of those features, and the relative importance of acquiring each type of measurement. In particular, the tools we describe enable the following capabilities: identifying geologic units of interest, identifying regions of contact between units, identifying the presence and orientation of layering within rock units, identifying novel features, selecting diverse targets for measurement, and designing rasters or image mosaics to measure identified regions of interest. Armed with this set of tools, future rovers can be instructed by scientists to perform high-level characterizations of new regions during periods without communication between the rover and Earth. The measurements acquired during an initial, autonomous characterization can be used by scientists to carefully select several key locations for more detailed, resource-intensive analyses. By aiming to more fully utilize the rover's capabilities at all times, the technologies we have developed will help increase rover mission productivity even in the face of communication constraints.

Acknowledgements

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Copyright 2017 California Institute of Technology. U.S. Government sponsorship acknowledged.

References

Burl, M.; Thompson, D.; deGranville, C.; and Bornstein, B. 2016. Rockster: Onboard rock segmentation through edge regrouping. *Journal of Aerospace Information Systems*.

Chien, S.; Doubleday, J.; Thompson, D. R.; Wagstaff, K.; Bellardo, J.; Francis, C.; Baumgarten, E.; Williams, A.; Yee, E.; Stanton, E.; et al. 2016. Onboard autonomy on the Intelligent Payload EXperiment CubeSat mission. *Journal of Aerospace Information Systems* 1–9.

Estlin, T.; Bornstein, B.; Gaines, D.; Anderson, R.; Thompson, D.; Burl, M.; Castaño, R.; and Judd, M. 2012. AEGIS automated targeting for the MER Opportunity rover. *ACM Transactions on Intelligent Systems and Technology* 3(3).

Francis, R.; Estlin, T.; Gaines, D.; Doran, G.; Gasnault, O.; Johnstone, S.; Montaño, S.; Mousset, V.; Verma, V.; Bornstein, B.; et al. 2016. AEGIS intelligent targeting deployed for the Curiosity rover's ChemCam instrument. In *Lunar and Planetary Science Conference*, volume 47.

Francis, R.; Gaines, D.; and Osinski, G. 2017. Advanced rover science autonomy experiments in preparation for the Mars 2020 mission: Results from the 2016 CanMars Analogue Mission. In *Lunar and Planetary Science Conference*, volume 48.

Gaines, D.; Doran, G.; Justice, H.; Rabideau, G.; Schaffer, S.; Verma, V.; Wagstaff, K.; Vasavada, A.; Huffman, W.; Anderson, R.; Mackey, R.; and Estlin, T. 2016. Productivity challenges for Mars rover operations: A case study of Mars Science Laboratory operations. Technical Report D-97908, Jet Propulsion Laboratory.

Liu, F. T.; Ting, K. M.; and Zhou, Z.-H. 2008. Isolation forest. In *Eighth IEEE International Conference on Data Mining*, 413–422. IEEE.

Malin, M.; Bell III, J.; Cameron, J.; Dietrich, W.; Edgett, K.; Hallet, B.; Herkenhoff, K.; Lemmon, M.; Parker, T.; Sullivan, R.; et al. 2005. The Mast Cameras and Mars Descent Imager (MARDI) for the 2009 Mars Science Laboratory. In *36th Annual Lunar and Planetary Science Conference*.

Newsom, H.; Belgacem, I.; Jackson, R.; Ha, B.; Vaci, Z.; Wiens, R.; Frydenvang, J.; Gasda, P.; Lanza, N.; Clegg, S.; et al. 2016. The materials at an unconformity between the Murray and Stimson formations at Marias Pass, Gale Crater, Mars. In *Lunar and Planetary Science Conference*, volume 47.

Stack, K.; Grotzinger, J.; Gupta, S.; Kah, L.; Lewis, K.; McBride, M.; Minitti, M.; Rubin, D.; Schieber, J.; Sumner, D.; Thompson, L.; VanBeek, J.; Vasavada, A.; and Yingst, R. 2015. Sedimentology and stratigraphy of the Pahrump Hills outcrop, Lower Mount Sharp, Gale Crater, Mars. In *Lunar and Planetary Science Conference*, volume 46.

Thompson, D. R.; Abbey, W.; Allwood, A.; Bekker, D.; Bornstein, B.; Cabrol, N.; Castaño, R.; Estlin, T.; Fuchs, T.; and Wagstaff, K. 2012. Smart cameras for remote science survey. In *Proceedings of the International Symposium on Artificial Intelligence, Robotics, and Automation in Space*.

Wagstaff, K. L.; Thompson, D. R.; Abbey, W.; Allwood, A.; Bekker, D. L.; Cabrol, N. A.; Fuchs, T.; and Ortega, K. 2013. Smart, texture-sensitive instrument classification for in situ rock and layer analysis. *Geophysical Research Letters* 40(16):4188–4193.

Wagstaff, K.; Lanza, N. L.; and Wiens, R. C. 2014. Unusual ChemCam targets discovered automatically in Curiosity's first ninety sols in Gale Crater, Mars. *45th Lunar and Planetary Science Conference*.

Wiens, R. C.; Maurice, S.; Barraclough, B.; Saccoccio, M.; Barkley, W. C.; Bell III, J. F.; Bender, S.; Bernardin, J.; Blaney, D.; Blank, J.; et al. 2012. The ChemCam instrument suite on the Mars Science Laboratory (MSL) rover: Body unit and combined system tests. *Space Science Reviews* 170:167–227.