Onboard Clustering of Aerial Data for Improved Science Return

David Hayden[(1)], Steve Chien[(2)], David R. Thompson[(3)], Rebecca Castano[(4)]

*[(1)] Jet Propulsion Lab, California Institute of Technology**
[(2)]Jet Propulsion Lab, California Institute of Technology, steve.chien@jpl.nasa.gov
[(3)] Jet Propulsion Lab, California Institute of Technology, david.r.thompson@jpl.nasa.gov
[(4)] Jet Propulsion Lab, California Institute of Technology, rebecca.castano@jpl.nasa.gov
** Contact Information: dshayden@asu.edu*

ABSTRACT

Current and future remote space missions, such as the aerial exploration of Titan by an aerobot, have the potential for collecting more data than can be returned for human observation. To relieve this bottleneck, we develop sub-polynomial algorithms for autonomously clustering aerial imagery. Specifically, we construct an Earth-based aerial image dataset as a Titan analogue. We compare performances of k-means-based clusterings against expert clustering. Among 21 low-level image descriptors accounting for color, texture, temporal and spatial arrangement, 5 are found which allow for clusterings that more closely match that of an expert than do clusterings based on random or periodic sampling. Results show potential for allowing scientists to infer semantic content of all images by downlinking representative images from each cluster. This allows for more efficient use of downlink bandwidth, and therefore higher quality science return in remote space exploration.

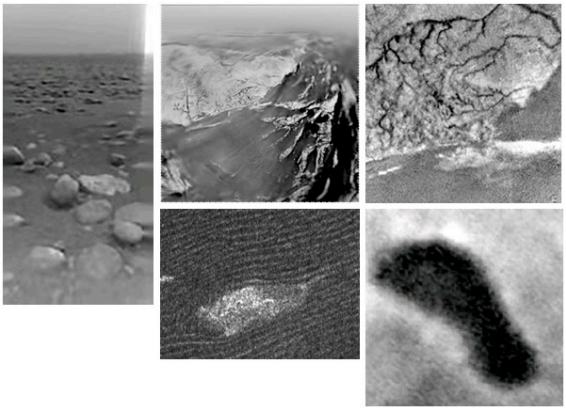

Figure 1: Cassini-Huygens images showing the diversity of Saturn's moon, Titan. Top-row images are from the Huygens probe and show a field containing frozen rocks with horizon, a hill etched by hydrocarbon rain, and part of a dried riverbed, respectively. Bottom-row images are from Cassini radar data and show sand dunes, and hydrocarbon liquid bodies, respectively.

1. INTRODUCTION

NASA's Solar System Exploration Strategic Roadmap [14] outlines the role of aerial vehicles in the future exploration of the solar system, particularly with respect to missions to Venus or Titan. In the case of Saturn's moon, Titan, an aerobot (blimp) would collect remotely-sensed data some 8km above ground level. It would be capable of circumnavigating the moon within a six month mission. The Cassini-Huygens mission has shown that Titan contains rich and varied landscapes (Figure 1), including smooth and rough terrain, sand dunes, ethane lakes, shorelines, craters, and possibly cryovolcanoes. Additionally, there is a significant cloud presence. With such Earth-like diversity, Titan is of great scientific interest.

Data yield for a Titan mission would be limited not by the rate of image acquisition, but rather by communications constraints. Communication with Earth would be subject to latencies that exceed two hours. Downlink bandwidth is expected to be 4500 bits/second, or 130Mbits / day assuming an 8 hour transmission window [7]. These considerations motivate autonomous methods of classifying aerial image data that could preselect the most scientifically meaningful data for return to Earth.

Previous work in onboard data understanding has focused on rover and satellite platforms. For the Mars Exploration Rovers (MER), science targets such as dust devils or rocks of specific size, albedo, and shape can be automatically recognized [3]. For the EO-1 Satellite, hazardous events such as fires, floods, and volcanic activities are detected and pertinent data downlinked. [5]. In both cases, targets can be detected without human direction, allowing for automatic data prioritization and improved science return.

Additional challenges exist for an aerobot; it would be in constant motion, but difficult to control due to unpredictable atmospheric currents. Processing would be shared between continuous autonomous control and data processing, but would be limited due to radiation hardening and energy constraints. A typical radiation-hardened processor used in space, the RAD6000, is clocked around 25MHz, has 128MB RAM, and can

Portions of this work were performed by the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration. Copyright 2009 © California Institute of Technology, all rights reserved.

maximally perform 35 MIPS—several orders of magnitude lower than modern computers. These limitations, combined with the diversity of surface features the aerobot might encounter, favors a computationally inexpensive, unsupervised approach that makes few assumptions about the image content the aerobot will encounter. Unsupervised methods have been proposed for selective data return applications in the rover domain. [2, 15]; they have also been widely used for image search and retrieval [4] and image sequence representation [10, 11] Clustering has been applied to aerial imagery [8], though not in an online fashion to our knowledge.

By clustering images as they are collected, an aerobot mission can analyze greater data volume. For example, the aerobot can downlink the *r* most exemplary images closest to each cluster centroid, providing a broad overview of the types of data collected. If some clusters prove to be consistently interesting then all images from those clusters can be prioritized for downlink. Finally, scientists can also opt to downlink images most dissimilar to the all others—outliers or anomalies. In summary, clustering enables selective data return based on a representative sample, a biased sample, or outliers. These options provide compelling alternatives to other data collection methods such as returning images collected at periodic intervals.

In this paper, we investigate the application of unsupervised classification for selective transmission of aerial image data in remote space exploration. We represent images in a metric space to compare their similarities. We identify specific image feature descriptors to encourage clusters based on semantic content such as presence of horizons, clouds, and water bodies. A broad survey of different image features suggests several that are both computationally efficient for spacecraft computers and relevant to the image categories identified by planetary scientists.

Section 2 begins by describing a small Earth-based aerial image dataset. Section 3 details the low-level, computationally inexpensive features we consider. These features account for color, texture, spatial arrangement, and time. Section 4 details both our algorithmic and manual (expert) clustering methods, as well as the metric with which we compare them. Sections 5 and 6 compare the effectiveness of different feature and clustering parameters. For our dataset, clustering based on several frequency-space features can more closely match an expert's clustering than either periodic or random sampling.

2. DATASET

We constructed a dataset of aerial imagery using a consumer-grade digital camera (Canon PowerShot SD850 IS) with resolution limited to 1600 x 1200. A total of 162 images were collected during a commercial airline flight from New York to Los Angeles. Images primarily contain shots dominated by sky, horizon, or undeveloped land. Some contain clouds, discernable water bodies, developed land, or have small portions of the plane's wing or window at one more edges. Figure 2 shows example images.

Factors such as varied terrain, presence of clouds, horizons, water bodies, and artifacts (e.g. the occasional window or wing obstruction) made this dataset a particularly appropriate Titan analogue.

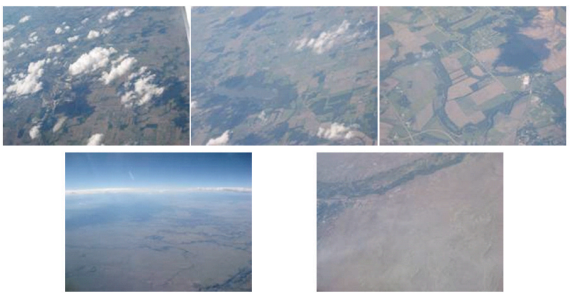

Figure 2: Exemplar images from a manual clustering performed by a planetary volcanologist who gave image clusters the names "more clouds", "rivers", "land", "horizon" and "desert".

3. FEATURES

We chose features to represent the color, texture, time, and spatial arrangement of each image. In order to reduce computational costs we favored simple features based on first- or second-order statistics that required little preprocessing.

We split the features into four themes: edge, color, frequency, and time. The edge and frequency features correlate with image texture, color captures basic color statistics, and time is an integer denoting the temporal order of image acquisition.

3A. Edge Features

Let I be an $m\,x\,d\,x\,3$ image, and I' be its $m\,x\,d$ grayscale. Let I'' be the resulting binary image from performing convolution with a Sobel operator on image I'. Let G_y, G_x be the $m\,x\,d$ matrices representing vertical and horizontal gradient responses, respectively. Let $\nabla L = \sqrt{G_y^2 + G_x^2}$ be the $m\,x\,d$ matrix representing gradient magnitudes, and $\theta = atan2(G_y, G_x)$ be the

$m\,x\,d$ matrix representing gradient orientations. Then the edge features are:

$F_0 = \frac{1}{md}\sum I'$ (edge density)

$F_1 = \frac{1}{md}\sum(\nabla \mathrm{L})$ (mean gradient magnitude)

$F_2 = H(\nabla L)$ (gradient magnitude entropy)

$F_3 = H(\theta)$ (gradient orientation entropy)

3B. Color Features

Let P_i be the $m\,x\,d$ matrix of pixels in band i of I. Then, for all P_i the color features are:

$F_{4..6} = \text{standard-deviation}(P_i)$

$F_{7..9} = \text{mean}(P_i)$

$F_{10..12} = \min(P_i)$

$F_{13..16} = \max(P_i)$

3C. Frequency Features

Let f be the resulting $m\,x\,d$ image after the 2D Fourier transform on I', and let $G(f) = |f|^2$ be the $m\,x\,d$ power spectrum of f. For the normalized power spectrum $N = \frac{G(f)}{\sum G(f)}$ let (m_x, m_y) be the coordinates of the pixel with the highest value. The frequency features are given by:

$F_{17} = m_x$ peak energy X

$F_{18} = m_y$ peak energy Y

$F_{19} = \sum_{x>0,y>0} N$ (quadrant-1 energy)

$F_{20} = \sum_{x<0,y>0} N$ (quadrant-2 energy)

These features were motivated by [12], which were found by a survey of 28 low-level frequency-space statistics to best discriminate a subset of the Brodatz textures.

3D. Time Features

Our dataset consists of an ordered sequence of 162 images. Let I_t represent this for each image. We generate an image feature based on the acquisition order:

$F_{21} = I_t$ (acquisition order)

3E. Spatial Features

In addition to accounting for texture, color, and time, we attempt to capture the spatial distribution of image features. Traditional approaches use image segmentation or region-growing methods to find objects or areas of uniform color or texture. These methods may be too expensive for aerobot applications.

Instead, we account for spatial arrangement by splitting each image into $n\,x\,n$ equal-sized subimages and collecting the features in each. Hence, for $n = 1$, we collect each desired feature once on the whole image, whereas for $n = 4$ we collect each feature in each of 16 subimages (except for F_{21}, which is never collected more than once). These subimage features are appended into an ordered vector whose dimensionality grows in proportion with the number of subimages.

4. CLUSTERING

We cluster using iterative k-means per the standard Lloyd's algorithm [13]. We initialize cluster centers using random datapoints and use a Euclidean distance metric to assign cluster membership. Image features were standardized prior to clustering.

Distances in the Euclidean space are sensitive to noisy or redundant dimensions. To account for these factors we employed a linear dimensionality reduction using Principal Component Analysis (PCA) with varying numbers of principal components.

3A. Expert Labelling

We elicited a manual clustering of our dataset from a planetary volcanologist to serve as a ground-truth standard for evaluation. The expert's only formal introduction to the task was a three minute introduction to simple graphical cluster selection software, and the following written prompt:

> Suppose that the following aerial images were taken of an environment for which we have little knowledge or data. Furthermore, suppose that you may not be able to receive all images. Please sort these images into 5 groups in such a way that if you could only receive a small number of images from each group, you could reasonably infer the content of the remaining images in that group.

We provided the expert a clustering tool based on a graphical drag-and-drop interface. As seen in Figure 3, the software contains three primary panes or windows. The leftmost window displays 160x120 thumbnails of unsorted images. The top-right contains a control panel that allows the user to name each cluster, while the bottom-right contains one or more scrollable subwindows that show the images within a cluster. The user drags thumbnails from, to, and between the

unsorted area and cluster windows. Thumbnails could be enlarged to full screen with a simple click.

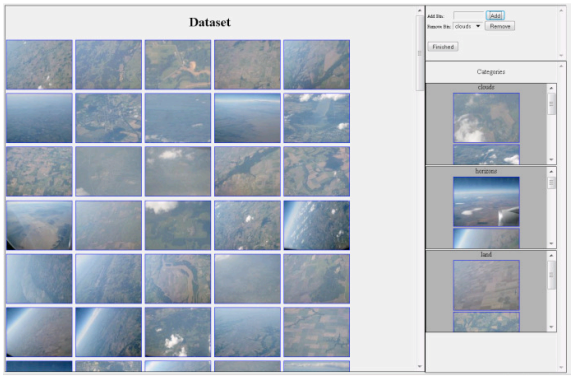

Figure 3: Custom software allows for rapid manual clustering of images. Users drag and drop images from the dataset (left) to bins they create and name (right).

We considered allowing the expert(s) to choose the number of clusters, but decided that this would provide too much flexibility given our intentionally vague prompt. This could create confusion about clusters' appropriate extent and roles. It is especially important since accepted methods of interpolating multiple, non-homogenous expert clusterings (e.g. [16]) are inadequate if scientists pursue multiple independent goals. We settled on 5 clusters for the simple reason that the number of clusters should be at least an order of magnitude less than the size of our dataset. The current study uses the data provided by a single expert. Upon interview, the expert felt that five was a mostly adequate number; six would have been ideal so that an outlier group could have been established. An exemplary image from each expert cluster is shown in Figure 2.

3B. Cluster Comparisons

We compare automatic clusterings against the expert standard using the information theoretic adjusted mutual information (AMI), which we briefly derive here.

Given dataset $S = \{s_1, s_2, \dots, s_N\}$, and two clusterings

$$U = \{U_1, U_2, \dots, U_R\} \quad (1)$$

$$V = \{V_1, V_2, \dots, V_C\} \quad (2)$$

where $\bigcap_{i=1}^{R} U_i = \emptyset$, and $\bigcup_{i=1}^{R} U_i = S$ (e.g. the U_i are a partitioning, or clustering, of S, and similarly for V). Then, the probability that a random data $s \in S$ is also contained in some cluster U_i is

$$P_u(i) = \frac{|U_i|}{N} \quad (3)$$

The probability that s is contained in some V_j is

$$P_v(j) = \frac{|V_j|}{N} \quad (4)$$

The joint probability that $s \in U_i$ and $s \in V_j$ is

$$P(i,j) = \frac{|U_i \cap V_j|}{N} \quad (5)$$

The mutual information is then defined as

$$MI(U,V) = \sum_{i=1}^{R} \sum_{j=1}^{C} P(i,j) \log\left(\frac{P(i,j)}{P_u(i)P_v(j)}\right) \quad (6)$$

Mutual information quantifies how much knowing about one clustering tells us about the other. Though it is symmetric and non-negative, it is not upper-bounded by a constant, and so is not useful as a general metric for comparing clusterings. Furthermore, Vinh et al. demonstrate that mutual information does not take a constant value when comparing random clusterings, and tends to grow with the number of clusters [17]. They use a hypergeometric model of randomness to derive an expected value for two random clusterings. This permits a correction similar to the Adjusted Rand Index [9] that ensures random clusterings produce a constant value, This correction yields the Adjusted Mutual Information (AMI):

$$AMI(U,V) = \frac{(MI(U,V) - E\{MI(U,V)\})}{\max\{H(U),H(V)\} - E\{MI(U,V)\}} \quad (7)$$

The entropies of clusterings U, V denote the uncertainty in a data point's cluster membership:

$$H(U) = -\sum_{i=1}^{R} P(i) \log P(i) \quad (8)$$

$$H(V) = -\sum_{j=1}^{C} P(j) log P(j) \quad (9)$$

The denominator in AMI both corrects for randomness and serves as a normalization, as otherwise $MI(U,V) \leq \min(H(U), H(V))$. Furthermore, $AMI(U,V) = 0$ only when equal to its expected value (e.g., that expected by two random clusterings), and $AMI(U,V) = 1$ when clusterings U, V are the same.

5. RESULTS

4A. Exhaustive Parameter Search

We begin with an exhaustive search over the parameter space to guide feature selection for further investigations. Specifically we cluster the dataset once for each possible combination of the four feature themes (edge, color, frequency, time). We consider each theme as the smallest unit (rather than each individual feature), which yields 15 potential combinations. Finally, we use

subimage decompositions of size $1, 4, 9,$ and 16, corresponding to $n = 1,2,3,4$.

We apply PCA with different dimensionalities to account for the curse of dimensionality and the potential for redundancy within, or interplay between, themes (e.g. the possibility that one edge feature and one color feature are most discriminative). We vary the number of principal components used from $1, 2, \dots, 19, 20, 24, 28, \dots, 96, 100$ or up to the dimensionality of the base feature set. Finally, we also vary the number of clusters, k, over 3,5,7,9,11.

In summary, we varied the following cluster parameters:

- The 15 theme combinations of {edge, color, frequency, time}.
- Using PCA vs. not using PCA.
- When using PCA, the number of principal components from $1, \dots, 100$.
- Number of clusters, k, from {3,5,7,9,11}.

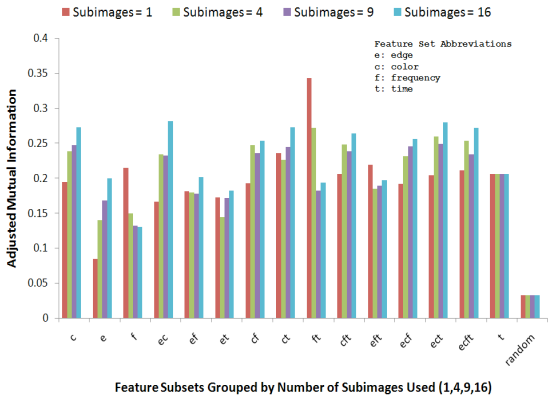

Figure 4: Best clustering performances among feature subsets over increasing numbers of subimages. Random clustering is also displayed.

Figure 5 summarizes the results in Figure 4 by only including the overall best performance among each feature theme subset. Results for random clustering are omitted to enhance visualization.

Clustering based on acquisition order alone ('t') yields a best AMI of approximately 0.21. We use this as a performance baseline since clustering based on acquisition order is equivalent to periodic sampling: each cluster will have $\frac{N}{k}$ images, and cluster centroids will be located at every $\left(\frac{N}{2k}\right)$ images in the sequence. Varying the number of clusters k, selects every i^{th} image for any i.

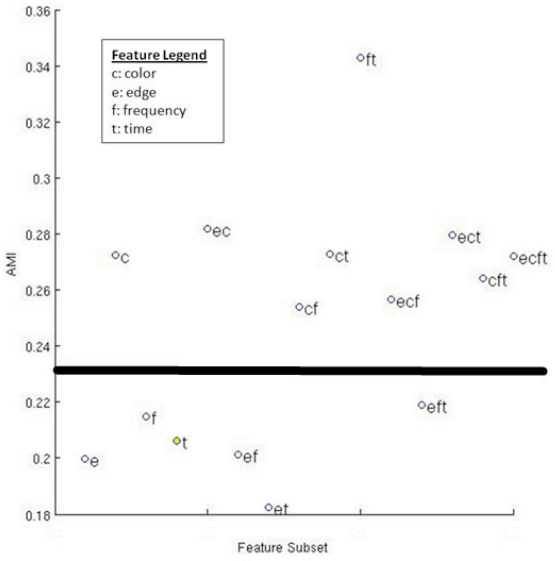

Figure 5: Best AMI for each subset of feature themes (edge, color, frequency, and time) in an exhaustive parameter search. The horizontal line separates feature themes whose performances are "close to" or below that of acquisition order.

We discard feature theme combinations whose best performances were near or below that obtained by acquisition order (0.21). Figure 5 illustrates this threshold with a horizontal line.

Increasing the number of subimages improves the clustering AMI for nearly all theme combinations. In general AMI does not exceed 0.26 for any subset of features unless collected under 16 subimages. The exception to both of these statements is the FT combination (frequency and time features). For frequency and frequency/time combinations a single subimage is most effective. For one subimage, the FT features yield the highest overall AMI of 0.343. We will consider this combination exclusively in the analysis that follows.

4B. Frequency-Space & Time Features

In further investigating clusterings with FT features, we first verify that larger numbers of subimages will statistically lower performance for our dataset. We fix $k = 5$ and use the ideal number of PCA components for 1, 4, 9, and 16 subimages. The ideal is determined by the highest mean AMI for 100 trials for all possible numbers of principal components up to the original dimensionality. Figure 6 displays these results.

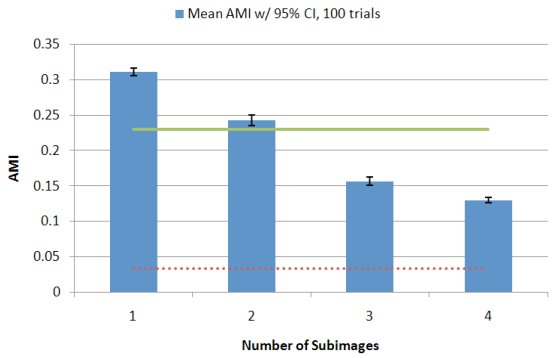

Figure 6: Clustering with FT Features, k=5, "ideal" value of PCA used for each subimage. The solid and dotted horizontal lines represent mean AMI over 100 trials of acquisition order and random clustering, respectively.

Figure 6 demonstrates that greater numbers of subimages fail to improve clustering performance of FT features, even with ideal numbers of PCA components.

We next consider the utility of PCA for one subimage. This is an appropriate question since the FT combination may not require all five features for optimal performance.

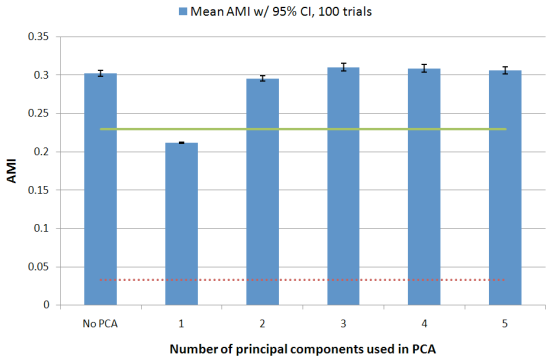

Figure 7: Clustering w/ FT features, k=5, 1 subimage, varying number of principal components used. The solid and dotted horizontal lines represent mean AMI over 100 trials of acquisition order and random clustering, respectively.

We see that there is no statistical significance between the base feature set and a compressed feature set of dimensionality 2 or greater (Figure 7). Designers can opt out of using PCA and suffer no detriment. An alternative interpretation is that there may be redundancy in our FT features that further testing could excise. We leave this to future work.

Finally, we vary the numbers of clusters within 1 subimage.

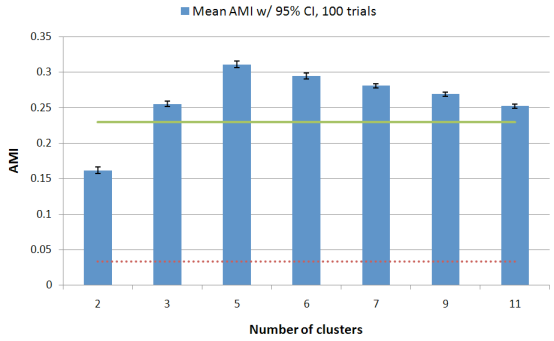

Figure 8: Clustering w/ FT features, 1 subimage, no PCA, varying number of clusters, k. The solid and dotted horizontal lines represent mean AMI over 100 trials of acquisition order and random clustering, respectively.

As expected, Figure 8 shows optimal performance when the number of k-means clusters matches that used in the expert's manual evaluation.

Finally, we compare the results of clustering with FT features against methods based on acquisition order and random clustering in Figure 9.

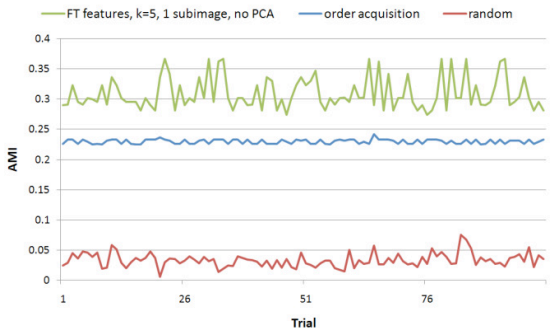

Figure 9: Comparison of 100 trials of clusterings with FT features against acquisition order and random clusterings.

Random clustering results in a low, fixed performance score and clustering with frequency and time features consistently outperforms clustering based on time alone.

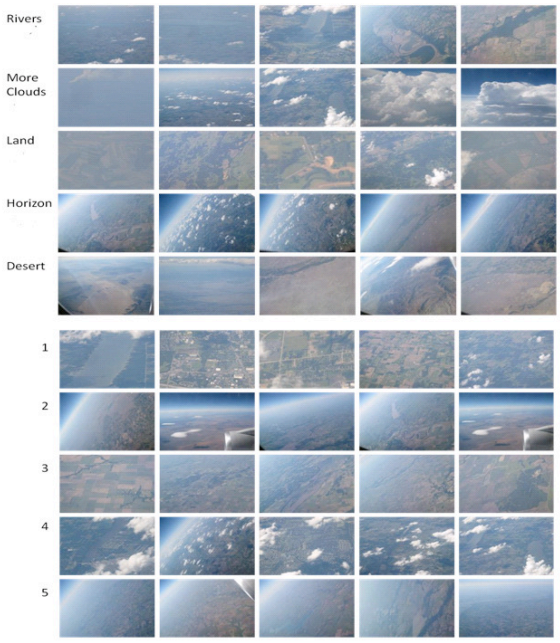

Figure 10: Comparison of exemplary images from an expert clustering (top) against k-means with FT features, 1 subimage, no PCA (bottom).

For qualitative observations, Figure 10 shows exemplary images from the expert clustering (top) compared with exemplars from k-means with FT features (below). We did not ask the expert to rank intra-cluster images, and so took a random sampling from each. Exemplars for automated clustering were chosen as the images closest to cluster centroids.

Exemplars for clusters 2 and 5 show images primarily containing horizons and so, together, corresponds well to the "Horizon" expert cluster. Exemplars for cluster 4 contain images with significant cloud presence, corresponding well to the "More Clouds" expert cluster. Clusters 1 and 3 contain an amalgam of ground-based imagery.

6. DISCUSSION

Our initial investigation revealed five features which outperform both random and acquisition order-based clustering. These features are relatively fast and simple to collect. The requisite Fourier transform is an $nlog\,n$ operation [6] while the normalization, search for maxima ($F_{17..18}$), and energy sums ($F_{19..20}$) are linear with respect to image size. PCA need not be used. With a fixed number of iterations, k-means runs in linear time. Therefore the total algorithmic complexity is sub-polynomial in image size and linear in the number of images.

The computational cost of the Fourier transform may be reduced if, in future work, it were found that these features worked as effectively on thumbnail-sized images or independent subimages.

It is well known that solutions provided by k-means are sensitive to initialization. In our results, we see that performance can vary by as much as 0.05 AMI, but it never underperforms clustering based on acquisition order. Even so, there has been significant work on clever initialization of k-means that tends to improve both performance and runtime (e.g. [1]). These should be considered for future work.

In our offline experiments features were standardized to a common scale. In an online environment, measures would have to be taken to find an appropriate scale. For the FT features, this would be simple as $F_{17..18}$ could be normalized, and $F_{19..20}$ already range between $0, \ldots, 1$.

Exploration of domain-specific features would also be useful. Determining methods that reliably form clusters for specific terrain (or its lower-level counterpart, texture) could aid specific science objectives. Beyond forming clusters based on terrain, it might also be interesting to consider forming clusters based on raw image quality. That is, can we easily set aside images which are marred with noise or sensor artifacts, or were collected with unfocused optics?

Finally, we are comparing against a single expert. The preliminary results should be verified against additional experts and datasets.

7. CONCLUSION

We have explored onboard, computationally inexpensive clustering for improving the science return of missions where more data can be collected than returned for human observation. Motivated by the potential for the aerial exploration of Titan via an aerobot, we have collected an Earth-based aerial image dataset and compared k-means clustering with that of a planetary volcanologist. Among 21 low-level features accounting for color, texture, spatial and temporal arrangement, 5 were found which more closely match the manual clustering of an expert than do clusterings formed by random or periodic sampling.

8. REFERENCES

[1] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, New Orleans,

Louisiana: Society for Industrial and Applied Mathematics, 2007, pp. 1035, 1027.

[2] R. Castano, T. Estlin, R.C. Anderson, D.M. Gaines, A. Castano, B. Bornstein, C. Chouinard, and M. Judd, "Oasis: Onboard autonomous science investigation system for opportunistic rover science: Research Articles," J. Field Robot., vol. 24, 2007, pp. 379-397.

[3] R. Castano, T. Estlin, D. Gaines, A. Castano, C. Chouinard, B. Bornstein, R. Anderson, S. Chien, A. Fukunaga, and M. Judd, "Opportunistic rover science: finding and reacting to rocks, clouds and dust devils," Aerospace Conference, 2006 IEEE, 2006, p. 16 pp.

[4] Y. Chen, J.Z. Wang, and R. Krovetz, "CLUE: Cluster-based Retrieval of Images by Unsupervised Learning," IEEE TRANSACTIONS ON IMAGE PROCESSING, vol. 14, 2003, p. 2005.

[5] S. Chien, B. Cichy, A. Davies, D. Tran, G. Rabideau, R. Castano, R. Sherwood, D. Mandl, S. Frye, S. Shulman, and others, "An autonomous earth-observing sensorweb," IEEE Intelligent Systems, 2005, pp. 16–24.

[6] J.W. Cooley and J.W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," Mathematics of Computation, vol. 19, Apr. 1965, pp. 297-301.

[7] J. Hall, V. Kerzhanovich, A. Yavrouian, J. Jones, C. White, B. Dudik, G. Plett, J. Mennella, and A. Elfes, "An aerobot for global in situ exploration of Titan," *Advances in Space Research*, vol. 37, 2006, pp. 2108-2119.

[8] K.R. Harvey and G.J.E. Hill, "Vegetation mapping of a tropical freshwater swamp in the Northern Territory, Australia: a comparison of aerial photography, Landsat TM and SPOT satellite imagery," International Journal of Remote Sensing, vol. 22, 2001, pp. 2911–2925.

[9] L. Hubert and P. Arabie, "Comparing partitions," Journal of Classification, vol. 2, Dec. 1985, pp. 193-218.

[10] L.S. Kennedy and M. Naaman, "Generating diverse and representative image search results for landmarks," Proceeding of the 17th international conference on World Wide Web, Beijing, China: ACM, 2008, pp. 297-306.

[11] S. Krishnamachari and M. Abdel-Mottaleb, "Image browsing using hierarchical clustering," Proceeding of 4 th IEEE Symposium on Computers and Communications (ISCC), 1999.

[12] S. Liu and M.E. Jernigan, "Texture analysis and discrimination in additive noise," Comput. Vision Graph. Image Process., vol. 49, 1990, pp. 52-67.

[13] S. Lloyd, "Least squares quantization in PCM," IEEE Transactions on Information Theory, vol. 28, 1982, pp. 129–137.

[14] NASA. "2006 solar system exploration roadmap" Technical report, NASA, 2006. Washington, DC.

[15] D. Thompson, T. Smith, and D. Wettergreen, "Information-optimal selective data return for autonomous rover traverse science and survey," *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, 2008, pp. 968-973.

[16] A. Vailaya, A. Jain, and Hong Jiang Zhang, "On image classification: city vs. landscape," Content-Based Access of Image and Video Libraries, 1998. Proceedings. IEEE Workshop on, 1998, pp. 3-8.

[17] N.X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: is a correction for chance necessary?," Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, Quebec, Canada: ACM, 2009, pp. 1073-1080.