Pose Sampling for Efficient Model-Based Recognition

Clark F. Olson

University of Washington Bothell, Computing and Software Systems
18115 Campus Way NE, Box 358534, Bothell, WA 98011-8246
http://faculty.washington.edu/cfolson

Abstract. In model-based object recognition and pose estimation, it is common for the set of extracted image features to be much larger than the set of object model features owing to clutter in the image. However, another class of recognition problems has a large model, but only a portion of the object is visible in the image, in which a small set of features can be extracted, most of which are salient. In this case, reducing the effective complexity of the object model is more important than the image clutter. We describe techniques to accomplish this by sampling the space of object positions. A subset of the object model is considered for each sampled pose. This reduces the complexity of the method from cubic to linear in the number of extracted features. We have integrated this technique into a system for recognizing craters on planetary bodies that operates in real-time.

1 Introduction

One of the failings of model-based object recognition is that the combinatorics of feature matching often do not allow efficient algorithms. For three-dimensional object recognition using point features, three feature matches between the model and the image are necessary to determine the object pose. Unless the features are so distinctive that matching is easy, this usually implies a computational complexity that is (at least) cubic the number of features (see, for example, [1,2,3]). Techniques using complex features [4,5,6], grouping [7,8,9], and virtual points [2] have been able to reduce this complexity in some cases, but no general method exists for such complexity reduction. Indexing can also be used to speed up recognition [10,11,12]. However, under the assumption that each feature set indexes a constant fraction of the database (owing to error and uncertainty), indexing provides a constant speedup, rather than a lower complexity [10,13].

We describe a method that improves the computational complexity for some cases. This method is valid for cases where the object model is large, but only part of it is visible in any image and at least a constant fraction of the features in the image can be expected to arise from the model. An example that is explored in this paper is the recognition of crater patterns on the surface of a planet.

The basic idea in this work is to (non-randomly) sample viewpoints of the model such that one of the sampled viewpoints is guaranteed to contain the

G. Bebis et al. (Eds.): ISVC 2007, Part II, LNCS 4842, pp. 781–790, 2007.

model features viewed in any image of the object. We combine this technique with an efficient model-based object recognition algorithm [3]. When the number of samples can be constrained to be linear in the number of model features and the number of salient features in each sample can be bounded, this yields an algorithm with computational complexity that is linear in both the number of image features and the number of model features.

Our pose sampling algorithm samples from a three degree-of-freedom space to determine sets of features that might be visible in order to solve full six degree-of-freedom object recognition. We do not need to sample from the full six dimensions, since the rotation around the camera axis does not change the features likely to be visible and out-of-plane rotations can usually be combined with translations in two degrees-of-freedom. Special cases may require sampling from more (or less) complex spaces. The set of samples is determined by propagating the pose covariance matrix (which allow an arbitrarily large search space) into the image space using the partial derivatives of the image coordinates with respect to the pose parameters.

We can apply similar ideas to problems where the roles of the image and model are reversed. For example, if a fraction of model is expected to appear in the image, and the image can be divided into (possibly overlapping) sets of features that can be examined separately to locate the model, then examination of these image sets can reduce the complexity of the recognition process.

Section 2 discusses previous research. Section 3 describes the pose sampling idea in more detail. We use this method in conjunction with efficient pose clustering, This combination of techniques is analyzed in Section 4. The methodology is applied to crater matching in Section 5 and the paper is concluded in Section 6.

2 Related Work

Our approach has a similar underlying philosophy to aspect graphs [14,15], where a finite set of qualitatively different views of an object are determined for use in recognizing the object. This work is different in several important ways. We do not attempt to enumerate all of the quantitatively different views. It is sufficient to sample the pose space finely enough that one of the samples has significant overlap with the input image. In addition, we can compute this set of samples (or views) efficiently at run-time, rather than using a precomputed list of the possible aspects. Finally, we concentrate on recognizing objects using discrete features that can be represented as points, rather than line drawings, as is typical in aspect graph methods.

Several other view-based object recognition methods have been proposed. Appearance-based methods using object views (for example, [16,17]) and those using linear combinations of views [18] operate under the assumption that an object can be represented using a finite set of views of the object. We use a similar assumption, but explicitly construct a set of views to cover the possible feature sets that could be visible.

Greenspan [19] uses a sampling technique for recognizing objects in range data. In this approach, the samples are taken within the framework of a tree search. The samples consist of locations in the sensed data that are hypothesized to arise from the presence of the object. Branches of the tree are pruned when the hypotheses become infeasible.

Peters [20] builds a static structure for view-based recognition using ideas from biological vision. The system learns an object representation from a set of input views. A subset of the views is selected to represent the overall appearance by analyzing which views are similar.

3 Pose Sampling

Our methodology samples from the space of poses of the camera, since each sample corresponds to a reduced set of model features that are visible from that camera position. For each sampled pose, the set of model features most likely to be detected are determined and used in a feature matching process. Not every sample from the pose space will produce correct results. However, we can cover the pose space with samples in such a way that all portions of the model that may be visible are considered in the matching process. Success, thus, should occur during one of the trials, if it would have occurred when considering the complete set of model features at the same time.

It is important to note that, even when we are considering a full six degree-of-freedom pose space, we do not need to sample from all six. Rotation around the camera axis will not change the features most likely to be visible in the image. Similarly, out-of-plane rotation and translation cause similar changes in set of the features that are likely to be visible (for moderate rotations). Therefore, unless we are considering large out-of-plane rotations, we can sample from a three-dimensional pose space (translations) to cover the necessary sets of model features to ensure recognition.

For most objects, three degrees-of-freedom are sufficient. If large rotations are possible, then we should instead sample the viewing sphere (2 degrees-of-freedom) and the distance from the object. For very large objects (or those for which the distance from the camera is very small), it may not be acceptable to conflate out-of-plane rotation and translation in the sampling. In this case, a five degree-of-freedom space must be sampled.

We define a grid for sampling in the translational pose space by considering the transverse motion (x and y in the camera reference frame) separately from the forward motion (z), since forward motion has a very different effect on the image than motion perpendicular to the viewing direction.

Knowledge about the camera position is represented by a pose estimate p (combining a translation t for the position and a quaternion q for the orientation) and a covariance matrix C in the camera reference frame. While any bounding volume in the pose space could be used, the covariance representation lends itself well to analysis. It allows an arbitrarily large ellipsoidal search space. While our pose representation has seven parameters (three for the translation and four for the quaternion), only six are independent.

For the z component of our sampling grid, we bound the samples such that a fixed fraction of the variance is enclosed (for example, three standard deviations around the pose estimate). Within this region, samples are selected such that neighboring samples represent a scale change by a fixed value, such as $\sqrt{2}$.

Each sampled z-coordinate (in the camera frame of reference), yields a new position estimate (according to the covariances with this z value) and we are left with a 6×6 covariance matrix in the remaining parameters. For each of these distances, we propagate the covariance matrix into the image space by determining a bounding ellipse for the image location of the object point at the center of the image for the input pose estimate. From this ellipse, we can determine the range over which to sample the transverse translations.

Let $\hat{p}$ be the vector $[0\ p]$. This allows us to use quaternion multiplication to rotate the vector. We can convert a point in the global frame of reference into the camera frame using:

$$p' = q\hat{p}q^* + t. \tag{1}$$

For a camera with focal length f, the image coordinates of a point are:

$$\begin{bmatrix} i_x \\ i_y \end{bmatrix} = \begin{bmatrix} \frac{fp'_x}{p'_z} \\ \frac{fp'_y}{p'_z} \end{bmatrix} \tag{2}$$

We now wish to determine how far the covariance matrix allows the location at the center of the image (according to the input pose estimate) to move within a reasonable probability. This variation is then accommodated by appropriate sampling from the camera translations. We can propagate the covariance matrix into the image coordinates using linearization by computing the partial derivatives (Jacobian) of the image coordinates with respect to the pose (Eq. 2). These partial derivatives are given in Eq. (3). The error covariance in the image space is $C_i = JC_pJ^T$, where C_p is the covariance matrix of the remaining six parameters in the camera reference frame.

$$J = \begin{bmatrix} \frac{\delta i_x}{\delta t_x} & \frac{\delta i_x}{\delta t_y} & \frac{\delta i_x}{\delta q_0} & \frac{\delta i_x}{\delta q_1} & \frac{\delta i_x}{\delta q_2} & \frac{\delta i_x}{\delta q_3} \\ \\ \frac{\delta i_y}{\delta t_x} & \frac{\delta i_y}{\delta t_y} & \frac{\delta i_y}{\delta q_0} & \frac{\delta i_y}{\delta q_1} & \frac{\delta i_y}{\delta q_2} & \frac{\delta i_y}{\delta q_3} \end{bmatrix}^T =$$

$$2f \begin{bmatrix} \frac{1}{2p'_z} & 0 \\ 0 & \frac{1}{2p'_z} \\ \frac{p'_z(-q_3p_y+q_2p_z)-p'_x(-q_2p_x+q_1p_y)}{{p'_z}^2} & \frac{p'_z(q_3p_x-q_1p_z)-p'_y(-q_2p_x+q_1p_y)}{{p'_z}^2} \\ \frac{p'_z(q_2p_y+q_3p_z)-p'_x(q_3p_x+q_0p_y-2q_1p_z)}{{p'_z}^2} & \frac{p'_z(q_2p_x-2q_1p_y-q_0p_z)-p'_y(q_3p_x+q_0p_y-2q_1p_z)}{{p'_z}^2} \\ \frac{p'_z(q_0p_z+q_1p_y-2q_2p_x)+p'_x(q_0p_x-q_3p_y+2q_2p_z)}{{p'_z}^2} & \frac{p'_z(q_1p_x+q_3p_z)-p'_y(-q_0p_x+q_3p_y-2q_2p_z)}{{p'_z}^2} \\ \frac{p'_z(-2q_3p_x-q_0p_y+q_1p_z)-p'_x(q_1p_x+q_2p_y)}{{p'_z}^2} & \frac{p'_z(q_0p_x-2q_3p_y+q_2p_z)-p'_y(-q_1p_x+q_2p_y)}{{p'_z}^2} \end{bmatrix} \tag{3}$$

The eigenvalues and eigenvectors of this covariance matrix indicate the shape of the area to sample from, with the eigenvectors being the axes of an ellipse and the square roots of the eigenvalues being the semi-axis lengths. We must now determine the spacing of the samples within these boundaries. Our strategy is to space the samples in a uniform grid aligned with the axes of the bounding ellipse such that the images that would be captured from neighboring samples overlap by 50 percent. This implies that, if the features are evenly distributed across the input image, one of the samples will contain a majority of the image features, even in the worst alignment with the sampling grid.

4 Efficient Pose Clustering

Our pose sampling technique has been combined with an efficient object recognition technique [3]. This method uses random sampling within the set of image features in order to develop a pose clustering algorithm that requires $O(mn^3)$ computation time, where m is the number of features in the model and n is the number of features in the image.

In previous analysis, it was assumed that some fraction of the model features must appear in the image in order for recognition to succeed. For the type of problem that we consider here, the model is large and the image covers a small portion of it. In addition, the image features are distinctive, with a significant fraction of them arising from the object model. Under these circumstances, the roles of the image and model are reversed in the analysis. We assume that at least some constant fraction of the image features arise from the model in order for recognition to succeed, but that only a small portion of the model may appear in the image. The number of model features that must appear in the image for recognition to succeed is not dependent on the size of the model. Following the analysis of [3], this implies a complexity of $O(m^3n)$ rather than $O(mn^3)$, since it is the image features that must be sampled, rather than the model features. Overall, $O(m^2)$ pairs of model features are sampled and each requires $O(mn)$ time prior to the application of the new pose sampling techniques.

The combination of pose sampling with this technique implies that the pose clustering technique must be applied multiple times (once for each of the sampled poses). This still results in improved efficiency, since the number of model features examined for each sampled pose is much smaller and the algorithm is cubic in this number.

The key to efficient operation is being able to set an upper bound on the number of model features that are examined for each pose. If this can be achieved, then the complexity for each of the sampled poses is reduced to $O(n)$, since the cubic portion is now limited by a constant. However, most sampled poses will not succeed and we must examine several of them. Since the number of model features that is examined for each pose is constant, we must examine $O(m)$ samples in order to ensure that we have considered the entire model. The overall complexity will therefore be $O(mn)$, if we can bound the number of model features examined in each pose sample by a constant and if the number of pose samples that are examined is $O(m)$.

We ensure that the number of model features examined for each sampled pose is constant by selecting only those that best meet predefined criteria (i.e., those most likely to be present and detected in the image given the sampled pose). Note also that the number of sampled poses in which each model feature is considered does not grow with the size of the model. This combined with the fact that each sample examines at least a constant number of model features (otherwise it can be discarded) implies that we examine $O(m)$ total samples.

To maintain $O(mn)$ efficiency, we must take care in the process by which the model features are selected for each sampled pose. Either the selection must be performed offline or an efficient algorithm for selecting them must be used online. Alternatively, each model feature can be considered for examination online for each sampled pose, but the algorithm becomes $O(m^2 + mn)$ in this case. In practice, this works well, since the constant on this term is small.

5 Crater Matching

We have applied pose sampling to the problem of recognizing a pattern of craters on a planet (or planetoid) as seen by a spacecraft orbiting (or descending to) the planet. In this application, we are able to simplify the problem, since the altitude of the spacecraft is well known from other sensors. This allows us to reduce the number of degrees-of-freedom in the space of poses that must be sampled from three to two. In addition, we have shown that many crater match sets can be eliminated efficiently using radius and orientation information [21].

For each pose that is sampled, we extract a set of the craters in the model that are most likely to be visible from that pose by examining those that are expected to be within the image boundaries and those that are of an appropriate size to be detected in the image. A set with bounded cardinality is extracted by ranking the craters according to these criteria.

Our first experiment used a crater model of the Eros asteroid that was extracted from images using a combination of manual and automatic processing at the Jet Propulsion Laboratory. See Fig. 1. Recognition was performed using a set of images collected by the Near Earth Asteroid Rendezvous (NEAR) mission [22]. Three images from this set can be seen in Fig. 1. Craters were first detected in these images using the method of Cheng *et al.* [23]. Results of the crater detection are shown in Fig. 1 (left column). The extracted craters, the crater model, and an inaccurate pose estimate were then input to the recognition algorithm described in this work. Figure 1 (right column) shows the locations where the visible craters in the model would appear according to the computed pose. The close alignment of the rendered craters with the craters visible in the image indicates that accurate pose estimation is achieved.

Our techniques found the same poses as detected in previous work [21] on this data with improved efficiency. With pose sampling, recognition required an average of 0.13 seconds on a Sun Blade™ 100 with a 500 MHz processor, a speedup of 10.2 over the case with no sampling.

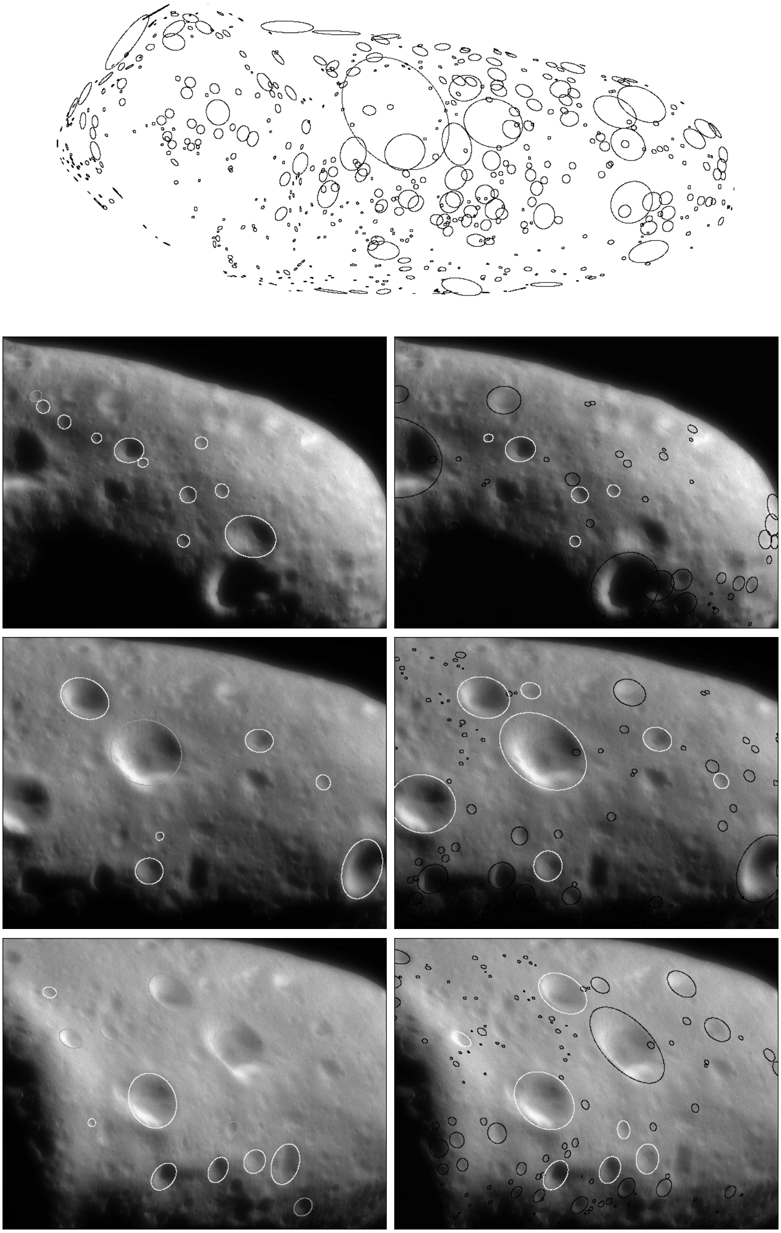

Fig. 1. Recognition of crater patterns on the Eros asteroid using images from the Near Earth Asteroid Rendezvous (NEAR) mission. (top) Rendering of a model of the craters on the Eros asteroid. (left) Craters extracted from NEAR images. (right) Recognized pose of crater model. Correctly matched craters are white. Unmatched craters are rendered in black according to the computed pose.

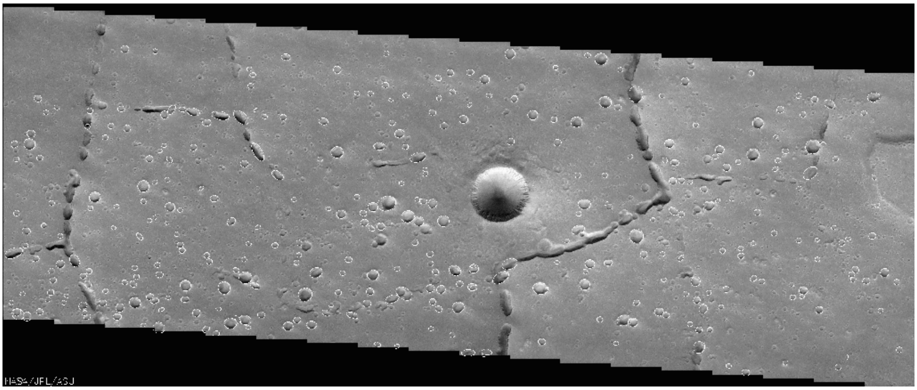

Fig. 2. Crater catalog extracted from Mars Odyssey data. Image courtesy of NASA/JPL/ASU.

Our second experiment examined an image of Mars captured by the THEMIS instrument [24] on the Mars Odyssey Orbiter [25]. The image shown in Fig. 2 shows a portion of the Mars surface with many craters. Crater detection [23] was applied to this image to create the crater model used in this experiment. Since the images in which recognition was performed for this experiment were resampled from the same image in which the crater detection was performed, these experiments are not satisfying as a measure of the efficacy of the recognition. However, our primary aim here is to demonstrate the improved efficiency of recognition, which these experiments are able to do.

Recognition experiments were performed with 280 image samples that cover the image in Fig 2. For examples in this set, we limited the number of features to the 10 strongest craters detected in the image and the 40 most likely craters to be visible for each pose. The correct qualitative result was found in each case, indicating that the sampling does not cause us to miss correct results that would be found without sampling. Four examples of the recognition results can be seen in Fig. 3. In addition, the pose sampling techniques resulted in a speedup by a factor of 9.02 with each image requiring 24.8 seconds on average with no input pose estimate. Experiments with the data set validate that the running time increases linearly with the number of features in the object model.

6 Summary

We have examined a new technique to improve the efficiency of model-based recognition for problems where the image covers a fraction of the object model, such as occurs in crater recognition on planetary bodies. Using this technique, we (non-randomly) sample from the space of poses of the object. For each pose, we extract the features that are mostly likely to be both visible and detected in the image and use these in an object recognition strategy based on pose clustering.

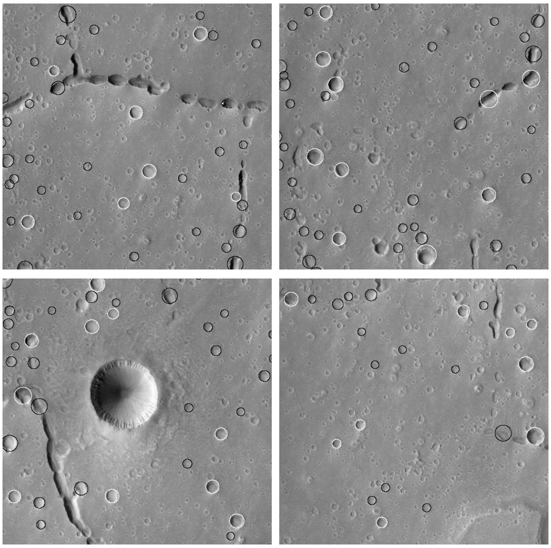

Fig. 3. Recognition examples using Mars Odyssey data. (Correctly matched craters are white. Unmatched craters are rendered in black according to the computed pose.)

When the samples are chosen appropriately, this results in a robust recognition algorithm that is much more efficient than examining all of the model features at once. A similar technique is applicable if the object is a small part of the image and the image can be divided into regions within which the object can appear.

Acknowledgments

We gratefully acknowledge funding of this work by the NASA Intelligent Systems Program. For the Eros crater model and the test images used in this work, we thank the Jet Propulsion Laboratory, the NEAR mission team, and the Mars Odyssey mission team.

References

1. Cass, T.A.: Polynomial-time geometric matching for object recognition. International Journal of Computer Vision 21, 37–61 (1997)
2. Huttenlocher, D.P., Ullman, S.: Recognizing solid objects by alignment with an image. International Journal of Computer Vision 5, 195–212 (1990)
3. Olson, C.F.: Efficient pose clustering using a randomized algorithm. International Journal of Computer Vision 23, 131–147 (1997)
4. Bolles, R.C., Cain, R.A.: Recognizing and locating partially visible objects: The local-feature-focus method. International Journal of Robotics Research 1, 57–82 (1982)

5. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60, 91–110 (2004)
6. Thompson, D.W., Mundy, J.L.: Three-dimensional model matching from an unconstrained viewpoint. In: Proceedings of the IEEE Conference on Robotics and Automation, vol. 1, pp. 208–220 (1987)
7. Havaldar, P., Medioni, G., Stein, F.: Perceptual grouping for generic recognition. International Journal of Computer Vision 20, 59–80 (1996)
8. Lowe, D.G.: Three-dimensional object recognition from single two-dimensional images. Artificial Intelligence 31, 355–395 (1987)
9. Olson, C.F.: Improving the generalized Hough transform through imperfect grouping. Image and Vision Computing 16, 627–634 (1998)
10. Clemens, D.T., Jacobs, D.W.: Space and time bounds on indexing 3-d models from 2-d images. IEEE Transactions on Pattern Analysis and Machine Intelligence 13, 1007–1017 (1991)
11. Flynn, P.J.: 3d object recognition using invariant feature indexing of interpretation tables. CVGIP: Image Understanding 55, 119–129 (1992)
12. Lamdan, Y., Schwartz, J.T., Wolfson, H.J.: Affine invariant model-based object recognition. IEEE Transactions on Robotics and Automation 6, 578–589 (1990)
13. Jacobs, D.W.: Matching 3-d models to 2-d images. International Journal of Computer Vision 21, 123–153 (1997)
14. Gigus, Z., Malik, J.: Computing the aspect graph for line drawings of polyhedral objects. IEEE Transactions on Pattern Analysis and Machine Intelligence 12, 113–122 (1990)
15. Kriegman, D.J., Ponce, J.: Computing exact aspect graphs of curved objects: Solids of revolution. International Journal of Computer Vision 5, 119–135 (1990)
16. Murase, H., Nayar, S.K.: Visual learning and recognition of 3-d objects from appearance. International Journal of Computer Vision 14, 5–24 (1995)
17. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience 3, 71–86 (1991)
18. Ullman, S., Basri, R.: Recognition by linear combinations of models. IEEE Transactions on Pattern Analysis and Machine Intelligence 13, 992–1006 (1991)
19. Greenspan, M.: The sample tree: A sequential hypothesis testing approach to 3D object recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 772–779 (1998)
20. Peters, G.: Efficient pose estimation using view-based object representation. Machine Vision and Applications 16, 59–63 (2004)
21. Olson, C.F.: Pose clustering guided by short interpretation trees. In: Proceedings of the 17th International Conference on Pattern Recognition, vol. 2, pp. 149–152 (2004)
22. http://near.jhuapl.edu/
23. Cheng, Y., Johnson, A.E., Matthies, L.H., Olson, C.F.: Optical landmark detection and matching for spacecraft navigation. In: Proceedings of the 13th AAS/AIAA Space Flight Mechanics Meeting (2003)
24. Christensen, P.R., Gorelick, N.S., Mehall, G.L., Murray, K.C.: (THEMIS public data releases) Planetary Data System node, Arizona State University, http://themis-data.asu.edu
25. http://mars.jpl.nasa.gov/odyssey/