OPTICAL NAVIGATION ALGORITHM PERFORMANCE

Cinnamon A. Wright[*], Andrew J. Liounis[†], and Benjamin W. Ashman[‡]

There is a wide variety of optical navigation (OpNav) techniques that can be used to extract observables from images of natural bodies. Each of these techniques has a number of strengths and weaknesses and domains where they are most applicable. In this paper, we compare the performance of some of the most commonly used OpNav techniques across a variety of orbital regimes and a variety of body types through the use of synthetic images. Specifically, we consider the techniques of analytic model fitting, phase corrected moment estimation, limb-scanning, ellipsoid matching, and cross correlation using synthetic images of a tri-axial ellipsoid, the asteroid Bennu, and the comet 67P/Churyumov-Gerasimenko. For each technique, regime, and body, we examine the overall accuracy and the type of information available. The resulting information provides a useful tool for understanding which techniques are best suited for a given image, as well as for understanding the relative performance of each technique.

INTRODUCTION

Optical Navigation (OpNav) refers to the set of techniques that are used to extract navigation observables from images of celestial bodies. It is an enabling capability for a wide range of missions including Lunar, interplanetary, small body, and even terrestrial, due to its ability to provide direct relative information between a spacecraft and its target. There are a number of different OpNav techniques that have been developed over the past 70 years that are still in use today. These techniques were developed with different computational constraints, different target body types, and different observables in mind, resulting in a wide variety of methods for extracting information from images.

Most modern OpNav techniques can be divided into three regimes depending on the apparent size of the body in the image: unresolved, resolved, and surface feature. In the unresolved regime, the target body appears as a point source in the image and the primary observable we can extract is simply a bearing to the center of the target body in the image. The resolved regime spans all the way from when the shape of the target body first becomes apparent in the image until the target body fills the field-of-view of the camera. There are two primary observables that we can extract from images in the resolved regime: bearing measurements to the center of the target body, and range measurements to the center of the target body. The surface feature regime begins when it becomes possible to identify individual features on the surface of the target body in the image; this

[*] Aerospace Engineer, Navigation and Mission Design Branch (Code 595), NASA Goddard Spaceflight Center, Greenbelt, MD 20771,U.S.A.

[†] Aerospace Engineer, Navigation and Mission Design Branch (Code 595), NASA Goddard Spaceflight Center, Greenbelt, MD 20771,U.S.A.

[‡] Aerospace Engineer, Navigation and Mission Design Branch (Code 595), NASA Goddard Spaceflight Center, Greenbelt, MD 20771,U.S.A.

typically overlaps the end of the resolved regime. In this regime, the primary observables are bearing measurements to individual surface features, and typically multiple features can be extracted from a single image.

In this paper, we compare the performance of some of the most common techniques from the unresolved and resolved regimes of OpNav across a range of body types and apparent body sizes in the image. In particular, we consider the techniques of analytic model fitting, phase corrected moment estimation, limb-scanning, ellipsoid matching, and cross correlation. We test the performance of these techniques by applying them to a set of synthetic images of three different target bodies which span a number of viewing conditions. The bodies we consider are a tri-axial ellipsoid (to represent regular bodies like large moons and planets), the asteroid Bennu (to represent semi-regular bodies that are still nearly entirely convex), and the comet 67P/Churyumov-Geasimenko (to represent very irregular bodies). The goal of this work is to provide a rough idea of where each algorithm performs best, and the relative accuracy that each technique can provide. We do not cover techniques from the surface feature realm in this work due to the large difference in the types of observables that are generated in this regime. For a discussion similar to this work that solely focuses on surface feature navigation, the reader is directed to Reference 1.

The rest of this work is organized as follows. First, we quickly review the background on each technique we are considering as a way to both introduce key terms and to indicate specific choices we have made in the implementation of the algorithms. We then discuss how we performed the analysis, including the source of our test images and our test setup. We continue by presenting the results of the analysis with comments. Finally, we conclude with a brief summary of the work and a discussion of future topics to be investigated.

BACKGROUND

In this section, we briefly review the particulars behind each algorithm that we tested in the analysis. This section is not intended to provide all of the details that are required to implement each algorithm, and instead strives to simply introduce the key concepts so that the reader may understand the overall idea of the algorithms. For a more in-depth discussion of the algorithms, the reader is directed to the references made in each section, or to Reference 2, which provides a general overview of the techniques.

Throughout this section and the rest of the paper, we will use the following nomenclature. The target location in the image is given by the sample and line locations s_c and l_c which are typically measured in units of pixels. The amount of light measured in each pixel in a digital image is represented as an array of "data numbers," or DN values.

Analytic Function Fitting

The first method we consider is analytic function fitting, which was developed as a way to estimate the sub-pixel center of objects in the image which are less than a couple of pixels across. In analytic function fitting, a point spread function (PSF) is fit to the DN values from an image to identify the center of the point source in the image.[2] The fit is generally performed using either linear least-squares estimation or iterative linearized least-squares estimation, where the point spread function is estimated to best minimize the residuals between the predicted DN values from the PSF and the measured DN values from the image. Typically, the PSF model used is either a 2D Gaussian function or a Lorentzian function, but any function which describes how a point source is blurred

by the lens of the camera can be used.

In this analysis, we use a 2D Gaussian as the point spread function to be fit to the data. We also linearize the function by taking the natural log, allowing us to estimate the parameters of the point spread function with a single linear least-squares estimate.* The PSF we fit is given by †

$$DN(s,l) = Ae^{-\left(a(x-x_0)^2 - 2*b*(x-x_0)*(y-y_0) + c*(y-y_0)^2\right)} \tag{1}$$

After taking the natural log and fitting data, the observed center of the PSF is $s_c = x_0$ and $l_c = y_0$.

Moment Algorithm

The next algorithm we consider is the moment algorithm, which is the most basic (and most efficient) algorithm covered. In the moment algorithm, we determine the center of brightness of the body in the image by using the moment algorithm given by

$$\begin{aligned} s_{COB} &= \frac{\sum_i \sum_j iDN_{i,j}}{\sum_i \sum_j DN_{i,j}} \\ l_{COB} &= \frac{\sum_i \sum_j jDN_{i,j}}{\sum_i \sum_j DN_{i,j}} \end{aligned} \tag{2}$$

where $\sum_i \sum_j$ sums over the pixels containing the body in the image. The range of pixels considered is typically determined by thresholding to ensure that the center of brightness estimate is not biased by the background or by other bright spots in the image. In this work, we use Otsu's method to automatically compute the threshold level needed to separate the target from the background, and we use connected components to remove extraneous bright spots caused by noise or stars.[3]

Once the center of brightness is found, it must be corrected for the effects caused by phasing to ensure that the measurement is to the center of the body, and not the center of the brightness. A variety of light-scattering models can be used to correct for this offset.[4] For this paper, Lambertian and Lommel-Seeliger sphere models[5] were used depending on the magnitude of the phase angle. The Lambertian offset is given by

$$d_{Lam} = R\frac{3\pi}{16}\frac{1+\cos\alpha}{(\pi-\alpha)\cot\alpha + 1} \approx R0.0065\alpha \tag{3}$$

and the Lommel-Seeliger is given by

$$d_{L-S} = R\frac{2}{3\pi}\frac{\sin\alpha + (\pi-\alpha)\cos\alpha}{\cot\frac{\alpha}{2} - \sin\frac{\alpha}{2} ln(\cot\frac{\alpha}{4})} \approx R0.0062\alpha \tag{4}$$

where R is the radius of the reference sphere in the image expressed in units of pixels, α is the phase angle in degrees, and d is the required correction to move the center of brightness estimate to the center of figure estimate we desire. The correction is applied in the direction of the sun projected into the image according to

$$\begin{pmatrix} s_c \\ l_c \end{pmatrix} = \begin{pmatrix} s_{COB} \\ l_{COB} \end{pmatrix} + d\begin{pmatrix} \hat{x}_s \\ \hat{y}_s \end{pmatrix} \tag{5}$$

*While linearizing the model enables us to perform the fit faster, it can cause issues because the natural log causes the tail of the PSF to be more highly weighted than the center of the PSF. In spite of this, as long as the body is small enough that its projection in the image is dominated by the PSF, it will be shown that the fit still performs well.

†With this we assume that the integrated DN values are represented by a Gaussian function and not the brightness function, as is done in Reference 2. While the formulation given in Reference 2 is more technically correct, using this formulation is much simpler and has produced excellent results in practice.

where $(\hat{x}_s \quad \hat{y}_s)^T$ is the unit vector in the direction of the incoming sunlight projected into the image. Figure 1 shows the geometry involved in this measurement type.

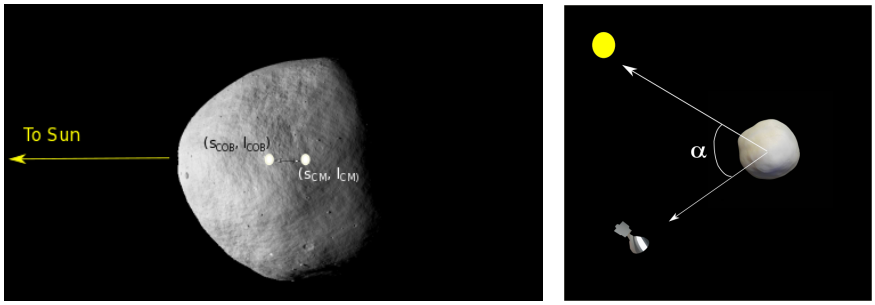

Figure 1. In the moment algorithm, a phase angle correction must be applied based on the phase angle between the illumination and observation directions.

Normalized Cross-Correlation

In normalized cross-correlation, the center-of-figure of the body in the image is determined by correlating a template of the predicted brightness values with the actual image. The correlation is performed according to

$$\rho(\Delta s, \Delta l) = \sum_i \sum_j \frac{\left(DN_{i,j} - \bar{DN}\right)\left(T_{i+\Delta s, j+\Delta l} - \bar{T}\right)}{n\sigma_{DN}\sigma_T} \tag{6}$$

where T is the template, $\bar{DN}$ is the average of the DN values over the pixels being considered, $\bar{T}$ is the average of the template values over the pixels being considered (accounting for the shift $(\Delta s \quad \Delta l)$), and σ_{DN} and σ_T are the standard deviations of the DN from the image and the template over the pixels being considered, respectively. This correlation is done for a number of $(\Delta s \quad \Delta l)$ and results in a correlation surface, the peak of which corresponds to the location of the center of the body in the image. We can further refine the peak by fitting a 2D parabaloid to the correlation surface in order to determine the sub-pixel peak location.

The template used to perform the cross correlation is generated by rendering the predicted view of the body using the *a priori* knowledge of the scene. The rendering can be done a number of ways, but in this analysis we use a single bounce ray tracer to compute the geometry seen by each pixel. The rays are traced from the pixels to the surface of the body. If the ray strikes the surface, then a new ray is traced from the intersection point to the sun to check if the point is illuminated or shadowed. If the ray is illuminated, then the brightness for that ray is computed using the McEwen bidirectional reflectance distribution function (BRDF).[6] Multiple rays are shot for each pixel and the resulting brightnesses are summed to ensure adequate sampling of the whole surface that is visible to each pixel.

Ellipse Matching

Ellipse matching is a relatively new technique developed for considering bodies that are well modeled by a tri-axial ellipsoid. It relies on the fact that when an ellipsoidal body is projected onto the 2D image plane, an ellipse is formed; this ellipse can be used to estimate the camera position

relative to the body. A detailed explanation can be found in Reference 7, but the solution to this conic section problem is described briefly here.

Points $\mathbf{p}$ on the surface of the ellipsoid satisfy

$$\mathbf{p}^T\mathbf{A}\mathbf{p} = 0, \tag{7}$$

where $\mathbf{A}$ is a 3x3 symmetric positive-definite matrix describing the shape of the body. When viewed from a relative position $\mathbf{r}$, the lens optical center and body limb form a cone, such that any ray $\mathbf{s}_i$ belonging to the cone satisfies

$$\mathbf{s}_i^T\left[\mathbf{A}\mathbf{r}\mathbf{r}^T\mathbf{A} - \left(\mathbf{r}^T\mathbf{A}\mathbf{r} - 1\right)\mathbf{A}\right]\mathbf{s}_i = 0. \tag{8}$$

Using matrix factorization, it is possible to transform this problem into one with a direct analytical solution. In the Cholesky factorized space, for instance, an arbitrary triaxial ellipsoid is a unit sphere, greatly simplifying the cone constraint problem in Equation (8). Consider a factorized space in which $\mathbf{A} = \mathbf{B}^T\mathbf{B}$, $\bar{\mathbf{s}}_i = \mathbf{B}\mathbf{s}_i$, and $\bar{\mathbf{s}}'_i = \bar{\mathbf{s}}_i/||\bar{\mathbf{s}}_i||$, where $||\cdot||$ is the vector norm. A number of factorizations are possible, but ultimately the problem is re-framed as

$$\begin{bmatrix} \bar{\mathbf{s}}'_1 \\ \bar{\mathbf{s}}'_2 \\ \vdots \\ \bar{\mathbf{s}}'_m \end{bmatrix} \mathbf{n} = \mathbf{1}_{mx1} \tag{9}$$

for m horizon measurements.[7] The full details of this transformation and the geometric meaning of $\mathbf{n}$ are provided in Reference 8 for the case of Cholesky factorization. After determining the least-squares solution for $\mathbf{n}$ from Equation (9), the camera position can be estimated

$$\hat{\mathbf{r}} = -(\mathbf{n}^T\mathbf{n} - 1)^{-(1/2)}\mathbf{B}^{-1}\mathbf{n}. \tag{10}$$

Limb-Scanning

In limb-scanning, limb points are identified in an image and matched with a corresponding surface point based on the *a priori* knowledge of the relative position and orientation between the camera and the target body. Given these 2D–3D correspondences, a least-squares estimation routine can be used to estimate an update to the location of the center-of-figure of the target body in the image.[2]

Traditionally, the image limb points have been extracted from the image by using a 1D cross correlation between the extracted image intensities and the predicted image intensities based off of the shape model and an intensity function along a predefined scan line.[2] In this work, however, we have used a more modern image processing approach where a sub-pixel edge-detection routine is used to identify the limb points in the image. Using a sub-pixel edge detector for extracting limb points is generally faster and more robust to errors in the *a priori* knowledge of the relative position and orientation between the camera and target body than using the traditional limb-scanning approach. In particular, we have used the Partial Area Effect edge detection method to extract the limb points from the image.[9]

We consider two different methods. For bodies that are well modeled by tri-axial ellipsoids (such as most moons and planets), the limb point along any scan line is given by a set of 3 equations

with 3 unknowns which has an analytic solution. For bodies that are not well modeled by tri-axial ellipsoids (such as asteroids and comets), the shape model generally takes the form of a triangular mesh that represents the surface. In this case, an analytic solution does not exist for a limb point along a given scan-line, and instead an iterative, numerical, binary search, ray tracing technique must be used. Both of these processes are described in detail in Reference 10.

Given the predicted and extracted limb points, an update to the body center can be estimated. Traditionally, the predicted limb points are just shifted in the image so that the residuals are minimized. In this work, however, we estimate the full 3D relative position between the camera and the target body, which best minimizes the residuals between the predicted and extracted limb points. This can be done for both tri-axial ellipsoids and irregular bodies. The required Jacobian matrices to perform this estimation are outside of the scope of this work and instead the reader is directed to Reference 10 for a more in-depth discussion.

METHODOLOGY

The goal of this paper is to demonstrate the capabilities of the OpNav techniques discussed in the previous section in various scenarios. This was done by applying each technique to a number of synthetic images generated from synthetic trajectories and shape models. The following subsections discuss the image generation and the application of the techniques.

Synthetic Image Generation

We generated synthetic images using the Geomod geometry modeling tool and the physical illumination and rendering system Phillum, a stochastic ray tracer. Both were developed at GSFC as part of the Freespace simulation environment. We simulated the images using a number of synthetic trajectories that were developed for testing by the OSIRIS-REx flight dynamics team. We specifically chose the synthetic trajectories to cover all the way from early approach, when the target body is still unresolved in the image, into orbit, when the target body nearly fills the entire field of view. Specifically, the apparent diameter of the target in the images spans from < 1 to 972 pixels. In addition, we used three different bodies as the target body for each image in the trajectories. These bodies were (a) a tri-axial ellipsoid with semi-axes lengths of $\begin{bmatrix} 0.25 & 0.3 & 0.15 \end{bmatrix}$ km, (b) a synthetic model of the 0.5 km diameter asteroid Bennu generated by the Altimetry Working Group at the OSIRIS-REx Science Processing and Operation Center with global 1 cm terrain[11] based on an Earth-based detailed radar and light curve observation of Bennu,[12] and (c) a model of the comet 67P/Churyumov-Gerasimenko generated from Rosetta imagery using Stereophotoclinometry and scaled by 0.14815 in each axis to make it similar in size to Bennu.[13] We also generated the images using three different cameras that were modeled to resemble the OSIRIS-REx PolyCam, MapCam, and NavCam cameras. Figures 2, 3, and 4 show a few sample renderings of each of these bodies.

Analysis

We processed the synthetic images using the Goddard Image Analysis and Navigation Tool (GIANT). GIANT is a high-fidelity OpNav tool that is capable of performing precision OpNav for highly challenging missions like OSIRIS-REx but is flexible enough to support rapid development, testing, and analysis. It incorporates a single bounce ray-tracer along with state-of-the-art image processing to enable autonomous use of all flavors of OpNav.

For each technique, where applicable, truth knowledge and truth models were used in order to

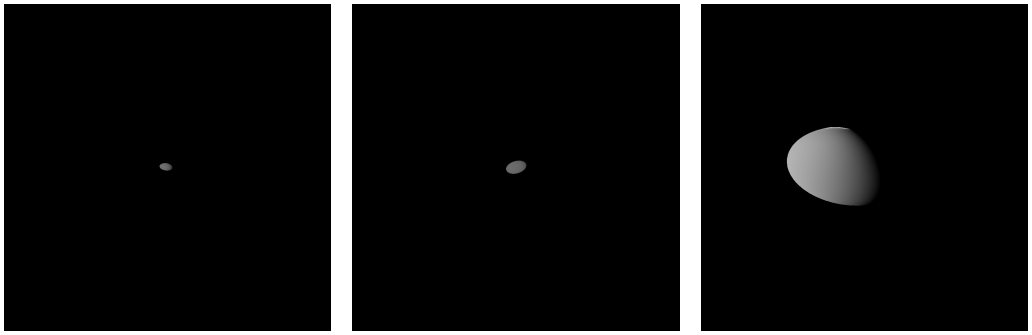

Figure 2. Sample synthetic images of the tri-axial ellipsoid.

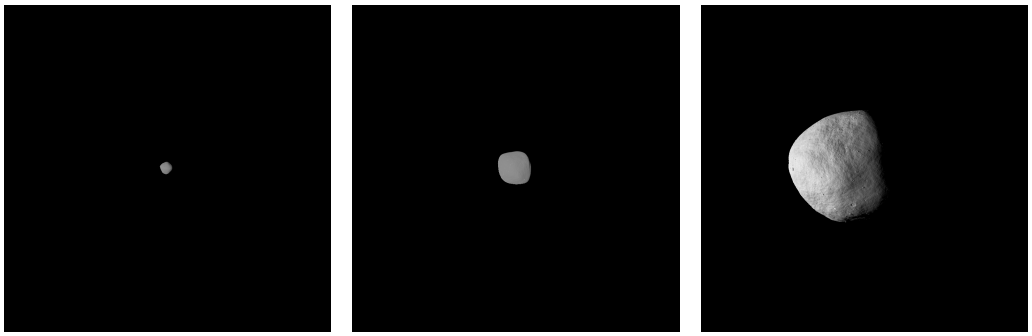

Figure 3. Sample synthetic images of the asteroid Bennu.

demonstrate the theoretical best-case capabilities of the techniques. In addition, each technique was applied to every image, although in some of the regimes, some of the techniques fail (like analytic function fitting). The results of each technique for each image were stored (either the 2D center-of-figure estimate of the body in the image or the 3D relative position between the camera and the target body) to be compared to truth. In addition, the apparent diameter of the body in each image was computed using the camera model, the range to the body, and an assumed maximum equatorial radius of 570 meters. All of the images were processed autonomously.

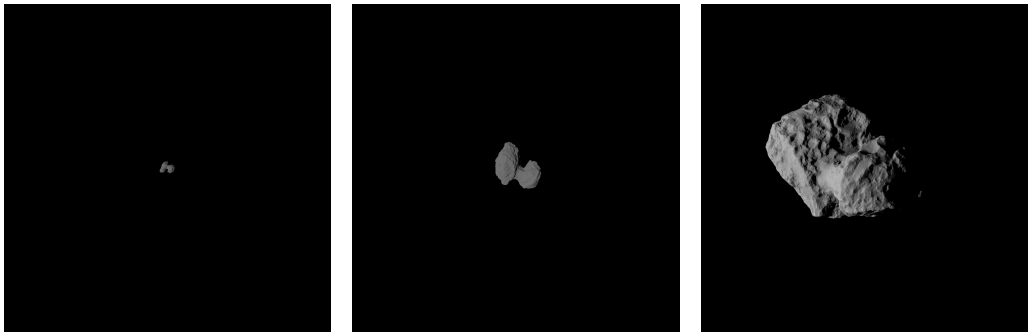

Figure 4. Sample synthetic images of the comet 67P/Churyumov-Gerasimenko.

RESULTS

The offsets between the "true" target center locations and the locations estimated by the five techniques are shown in Figures 5 - 11. Each plot shows the same data, but cropped in a different way to more clearly demonstrate the performance of the different techniques. Observations about the performance for each technique are made in the following subsections. Figure 6 shows a case where the combination of the non-uniform shape and lighting conditions causes the body to appear as two separate bodies. This can cause outliers for some of the methods, as will be discussed.

Analytic Function Fitting

The analytic function fitting performs very well (sub-pixel) until the target becomes larger than about 10 pixels for all body types (shown most clearly in Figure 7). Once the target becomes larger than about 20 pixels in the image, then the analytic function fitting begins to fail to the point where no estimate is even produced. When the body is less than 5 pixels across, this technique provides the best performance out of all of the techniques considered.

Moment Algorithm

The moment algorithm performs fairly well, especially for the ellipse and Bennu until the body is about 30 pixels in size. As the body takes up more of the image, the moment algorithm results begin to degrade. This divergence is increased in the ellipsoid and 67P due to their very non-spherical shapes. The moment algorithm never provides the best performance for any regime being considered; however, it does provide good rough estimates that are cheap to compute, particularly when the body is smaller and close to a sphere in shape.

Ellipse Matching

As expected, during early approach when the target is too small to be approximated as an ellipse, large offsets in the target location are seen, shown most clearly in Figure 9. The ellipse matching algorithm performs very well for the ellipsoid, once the body becomes greater than about 20 pixels across, and almost as well for Bennu up until the body becomes about 200 pixels across. For Bennu, once the body becomes greater than about 200 pixels across, then the surface topography begins to dominate the overall shape of the limb, causing the ellipsoid approximation to break down and the technique to begin to deteriorate. 67P/Churyumov-Gerasimenko is not as well approximated as an ellipse, so the results quickly deteriorate even when the body is still very small. For the ellipsoid, this technique produces some of the best results once the body grows larger than about 20 pixels across.

Limb Scanning

The limb-scanning method begins to work well once the body exceeds about 15 pixels for all body types, until it is so close that the limb is out of the field of view, shown most clearly in Figure 11. In this regime, it consistently provides some of the best results out of the techniques that were considered. There are a few outlier cases where the predicted and extracted limb points were not paired correctly, particularly for 67P, but overall, the results are excellent. For the ellipsoid, this technique produces nearly identical results to the ellipse matching technique, as expected.

Normalized Cross-Correlation

The cross-correlation technique is consistently one of the best performers for all bodies and all apparent diameters, earning well its reputation as the workhorse OpNav technique. The offsets during approach are slightly larger for the ellipse. This may be due to the uniformity of the body and lack of variation in terrain. Also, in the ellipse case there is a small grouping of outliers (around 2 pixels) as the target body becomes larger (most apparent in Figure 10). This is due to an unusual bright spot along the limb that was an artifact of the rendering; this can be seen in Figure 2.

CONCLUSION

The normalized cross-correlation performs well for all bodies and all regimes. The analytic function fitting works well if a body is less than 10 pixels in apparent diameter but breaks down above that size. The moment algorithm is the simplest to implement and performs adequately until the target takes up about 30 pixels (as long as the body is roughly spherical), at which point the performance begins to slowly degrade. The limb-scanning method performs well for all body types once the body exceeds about 15 pixels in apparent diameter. The ellipse matching performs very well for the ellipsoid once the body exceeds about 15 pixels in diameter, but the results degrade quickly for 67P and, above 200 pixels, for Bennu.

For stars or celestial bodies that are very far away and take up less than a few pixels, analytic function fitting and the moment algorithm both work very well. For a body that is very close to an ellipsoid, ellipse matching or limb-scanning both produce excellent center finding results and also produce estimates about the range to the body. For irregular bodies, limb-scanning performs very well in center-finding and still produces a range estimate to the body. Normalized cross-correlation consistently produces some of the best center finding results regardless of body type or apparent size.

These are preliminary results produced with synthetic imagery. Future work includes testing these methods on actual imagery and performing a Monte-Carlo analysis varying the *a priori* knowledge to see how the algorithms perform when error is involved.

ACKNOWLEDGMENT

The authors would like to thank Olivier Barnouin, Eric Palmer, Michael Daly, Robert Gaskell, and Catherine Johnson of the ALTWG for the excellent work they performed in generating the surface models used to simulate the images of the asteroid Bennu. We would also like to thank Robert Gaskell for producing the 67P/Chruyumov-Gerasimenko shape model. Finally, thanks to Kenneth Getzandanner and Joshua Lyzhoft of NASA Goddard and Natalie Liounis for their valuable feedback.

REFERENCES

[1] A. J. Liounis, C. Jackman, K. Getzandanner, and L. McCarth, "Sensitivity Analysis of Surface Feature Navigation Using Stereophotoclinometry," *27th AAS/AIAA Space Flight Mechanics Meeting*, San Antonio, TX, 2017.

[2] W. M. Owen Jr., "Methods of Optical Navigation," *21st Annual AAS/AIAA Space Flight Mechanics Meeting*, New Orleans, AAS, 2011.

[3] N. Otsu, "A Threshold Selection Method From Gray-Level Histograms," *IEEE Transactions On Systems, Man, and Cybernetics*, Vol. 9, No. 1, 1979, pp. 62–66.

[4] S. Bhaskaran, J. E. Riedel, and S. P. Synnott, "Autonomous optical navigation for intclplanctary missions," *Navigation and Flight Mechanics Section*.

[5] M. Kaasalainen and P. Tanga, "Photocentre offset in ultraprecise astrometry: Implications for barycentre determination and asteroid modelling," *Astronomy & Astrophysics*, Vol. 416, No. 1, 2004, pp. 367–373.

[6] A. S. McEwen, "Photometric Functions for Photoclinometry and Other Applications," *Icarus*, Vol. 92, No. 2, 1991, pp. 298 – 311, https://doi.org/10.1016/0019-1035(91)90053-V.

[7] J. A. Christian, "Accurate Planetary Limb Localization for Image-Based Spacecraft Navigation," *Journal of Spacecraft and Rockets*, 2017, pp. 1–23.

[8] J. A. Christian and S. B. Robinson, "Noniterative Horizon-Based Optical Navigation by Cholesky Factorization," *Journal of Guidance, Control, and Dynamics*, 2016, pp. 2755–2763.

[9] A. Trujillo-Pino, K. Krissian, M. Alemn-Flores, and D. Santana-Cedrs, "Accurate Subpixel Edge Location Based on Partial Area Effect," *Image and Vision Computing*, Vol. 31, No. 1, 2013, pp. 72 – 90, https://doi.org/10.1016/j.imavis.2012.10.005.

[10] A. J. Liounis, "Limb Based Optical Navigation for Irregular Bodies," *1st Annual RPI Workshop on Image-Based Modeling and Navigation for Space Applications*, Troy, NY, 2018.

[11] J. R. Weirich, E. E. Palmer, and R. W. Gaskell, "Testing of Stereophotoclinometry (SPC) Software in Support of the OSIRIS-REx Mission," *Lunar and Planetary Science Conference*, Vol. 47 of *Lunar and Planetary Science Conference*, Mar. 2016, p. 2777.

[12] M. C. Nolan, C. Magri, E. S. Howell, L. A. Benner, J. D. Giorgini, C. W. Hergenrother, R. S. Hudson, D. S. Lauretta, J.-L. Margot, S. J. Ostro, and D. J. Scheeres, "Shape Model and Surface Properties of the OSIRIS-REx Target Asteroid (101955) Bennu From Radar and Lightcurve Observations," *Icarus*, Vol. 226, No. 1, 2013, pp. 629 – 640, http://dx.doi.org/10.1016/j.icarus.2013.05.028.

[13] H. Sierks, C. Barbieri, P. L. Lamy, R. Rodrigo, D. Koschny, H. Rickman, H. Uwe Keller, J. Agarwal, M. F. AHearn, F. Angrilli, A.-T. Auger, M. Barucci, J.-L. Bertaux, I. Bertini, S. Besse, D. Bodewits, C. Capanna, G. Cremonese, V. Da Deppo, and M. Ptzold, "On the Nucleus Structure and Activity of Comet 67P/Churyumov-Gerasimenko," Vol. 347, 01 2015.

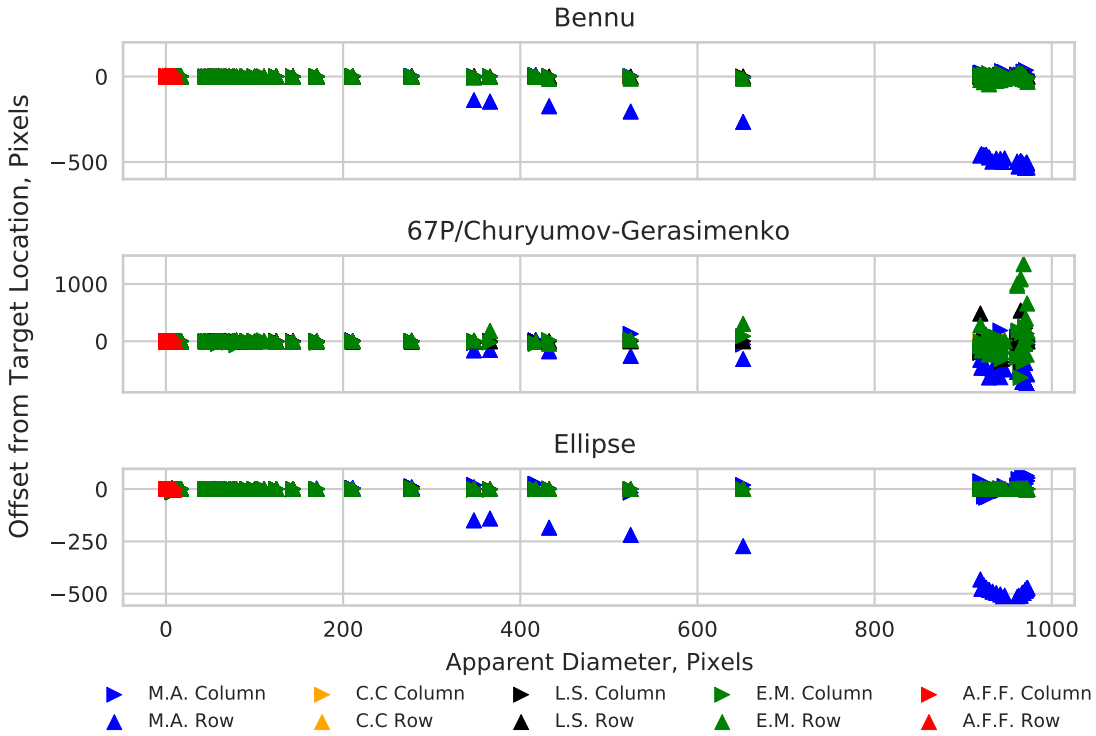

Figure 5. Center finding results for each body type and each apparent diameter considered. The different techniques are identified by the different colors shown in the legend. The top plot is for images of Bennu, the middle plot is for images of 67P/Churyumov-Gerasimenko, and the bottom plot is for images of a tri-axial ellipsoid. In the legend M.A. represents moment algorithm, C.C. represents cross-correlation, L.S. represents limb-scanning, E.M. represents ellipse matching, and A.F.F. represents analytic function fitting.

Figure 6. Image of 67P demonstrating how a non-uniform shape can appear to be two separate bodies.

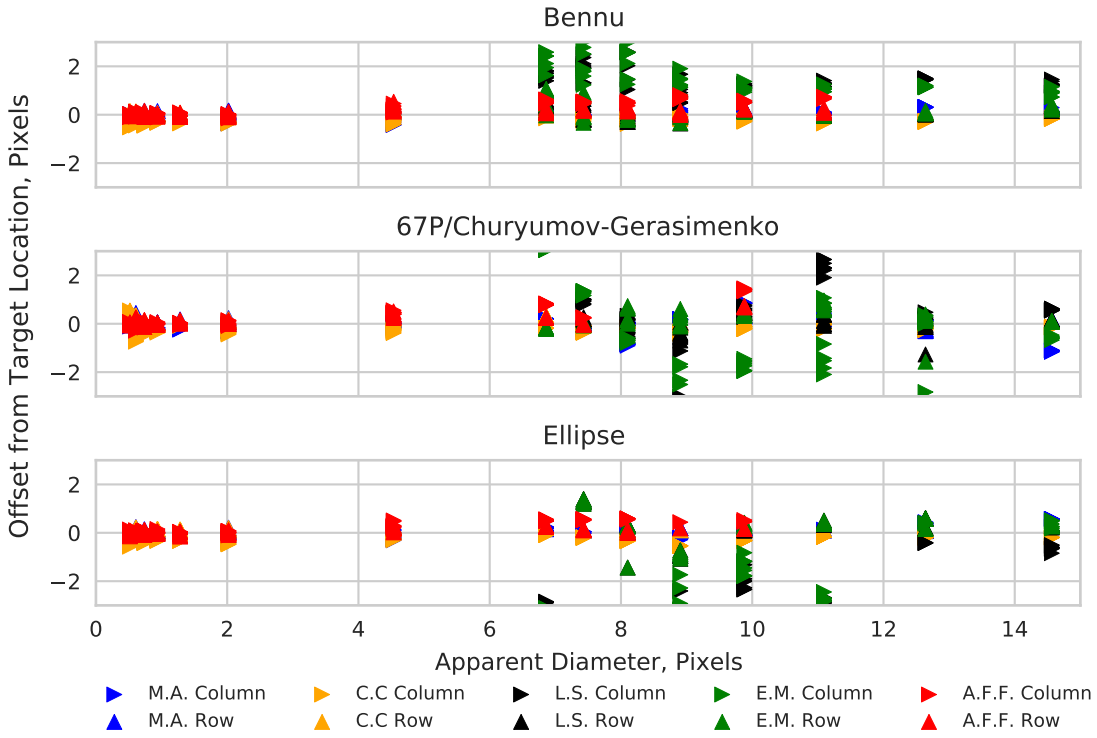

Figure 7. Center finding results cropped to errors less than 3 pixels and apparent diameter less than 15 pixels to demonstrate the analytic function fitting performance. The different techniques are identified by the different colors shown in the legend. The top plot is for images of Bennu, the middle plot is for images of 67P/Churyumov-Gerasimenko, and the bottom plot is for images of a tri-axial ellipsoid. In the legend M.A. represents moment algorithm, C.C. represents cross-correlation, L.S. represents limb-scanning, E.M. represents ellipse matching, and A.F.F. represents analytic function fitting.

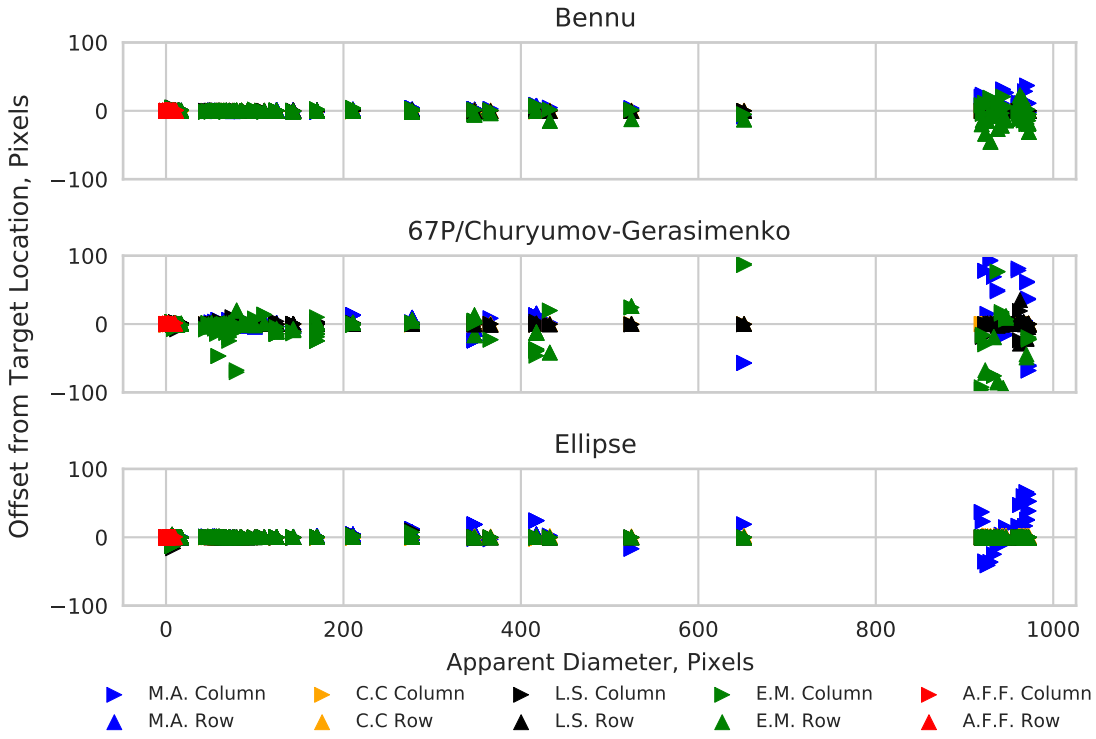

Figure 8. Center finding results for each body type and each apparent diameter considered cropped to errors of less than 100 pixels. The different techniques are identified by the different colors shown in the legend. The top plot is for images of Bennu, the middle plot is for images of 67P/Churyumov-Gerasimenko, and the bottom plot is for images of a tri-axial ellipsoid. In the legend M.A. represents moment algorithm, C.C. represents cross-correlation, L.S. represents limb-scanning, E.M. represents ellipse matching, and A.F.F. represents analytic function fitting.

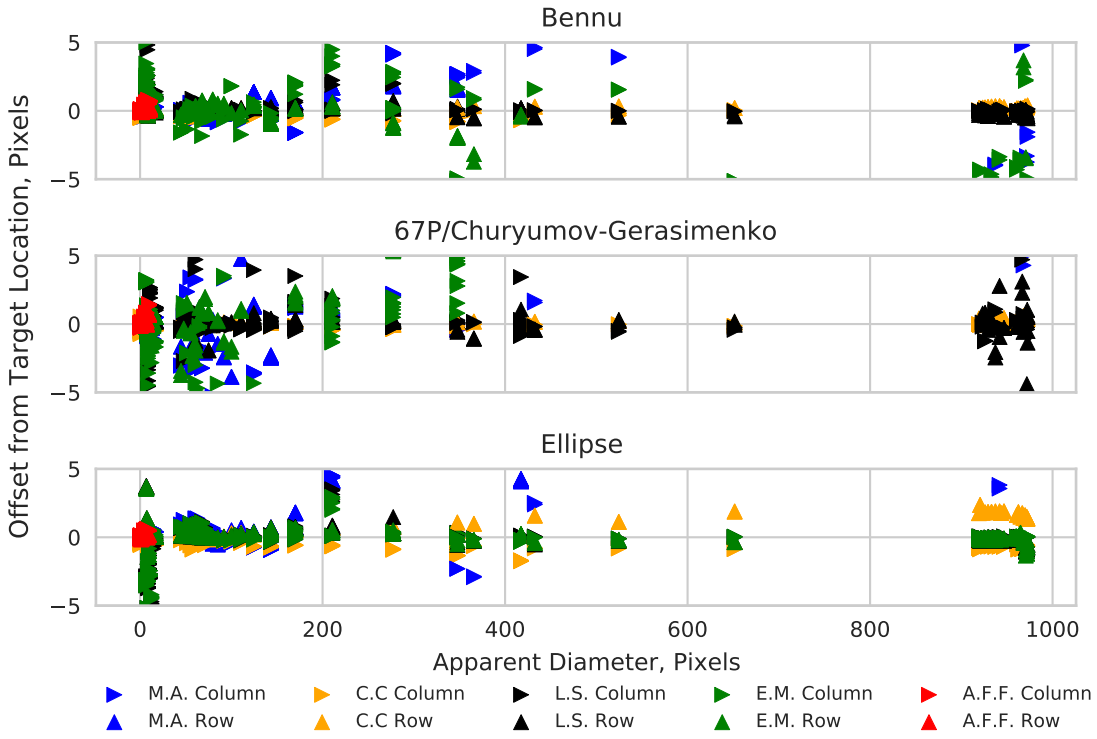

Figure 9. Center finding results for each body type and each apparent diameter considered cropped to errors of less than 5 pixels. The different techniques are identified by the different colors shown in the legend. The top plot is for images of Bennu, the middle plot is for images of 67P/Churyumov-Gerasimenko, and the bottom plot is for images of a tri-axial ellipsoid. In the legend M.A. represents moment algorithm, C.C. represents cross-correlation, L.S. represents limb-scanning, E.M. represents ellipse matching, and A.F.F. represents analytic function fitting.

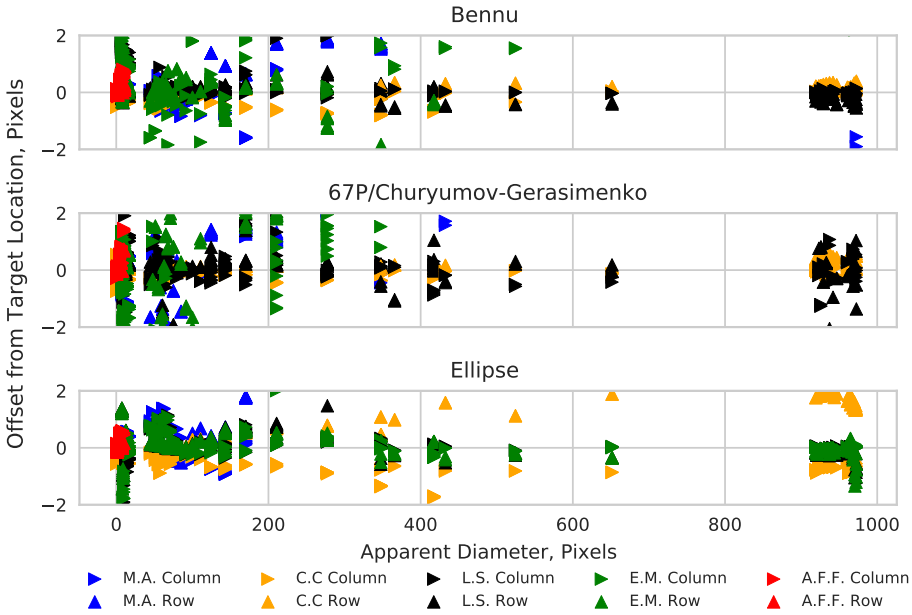

Figure 10. Center finding results for each body type and each apparent diameter considered cropped to errors of less than 2 pixels. The different techniques are identified by the different colors shown in the legend. The top plot is for images of Bennu, the middle plot is for images of 67P/Churyumov-Gerasimenko, and the bottom plot is for images of a tri-axial ellipsoid. In the legend M.A. represents moment algorithm, C.C. represents cross-correlation, L.S. represents limb-scanning, E.M. represents ellipse matching, and A.F.F. represents analytic function fitting.

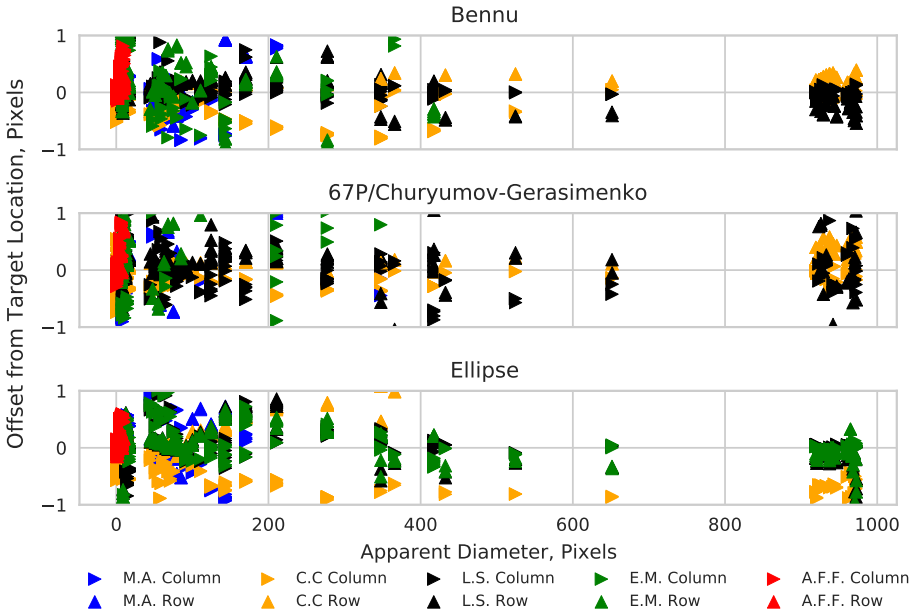

Figure 11. Center finding results for each body type and each apparent diameter considered cropped to errors of less than 1 pixel. The different techniques are identified by the different colors shown in the legend. The top plot is for images of Bennu, the middle plot is for images of 67P/Churyumov-Gerasimenko, and the bottom plot is for images of a tri-axial ellipsoid. In the legend M.A. represents moment algorithm, C.C. represents cross-correlation, L.S. represents limb-scanning, E.M. represents ellipse matching, and A.F.F. represents analytic function fitting.