Spacecraft Relative Navigation Using Appearance Matching and Sensor Fusion

Christopher R. McBryde, *Georgia Institute of Technology*
E. Glenn Lightsey, *Georgia Institute of Technology*

BIOGRAPHIES

Christopher R. McBryde is a PhD Candidate in the Daniel Guggenheim School of Aerospace Engineering at the Georgia Institute of Technology. His researches visual navigation technology for satellites under Dr. Glenn Lightsey. Christopher is a NASA Space Technology Research Fellow and has done worth through NSTRF at Ames Research Center and the Jet Propulsion Laboratory. Prior to his work at Georgia Tech, Christopher received his master's degree from the University of Texas at Austin, also working with Dr. Lightsey. His research contributed to two satellite missions by the Texas Spacecraft Laboratory: Bevo-2 and ARMADILLO.

E. Glenn Lightsey is a Professor in the Daniel Guggenheim School of Aerospace Engineering at the Georgia Institute of Technology. He previously worked at the University of Texas at Austin and NASA's Goddard Space Flight Center. His research program is focused on the technology of satellites, including: guidance, navigation, and control systems; attitude determination and control; formation flying, satellite swarms, and satellite networks; cooperative control; proximity operations and unmanned spacecraft rendezvous; space based Global Positioning System receivers; radionavigation; visual navigation; propulsion; satellite operations; and space systems engineering. At the University of Texas, he founded and directed the Texas Spacecraft Lab which built university satellites. He has authored and coauthored more than 120 technical publications. He is an AIAA Fellow, and he currently serves as Associate Editor-in-Chief of the *Journal of Small Satellites* and Associate Editor of the *AIAA Journal of Spacecraft and Rockets.*

ABSTRACT

In this research, the task of object recognition and relative navigation is accomplished by fusing visible spectrum and infrared images. The appearance matching technique is briefly explained and it is shown how it can be extended to infrared images. A series of tests are performed to demonstrate the object recognition and pose estimation capabilities of the system in the visible and infrared spectra. It is also shown how the fusion of both types of images can provide greater accuracy and robustness in relative navigation than either visual or infrared images alone. Additionally, a simulation environment software tool has been developed to facilitate the creation of training images and to perform software-in-the-loop verification.

INTRODUCTION

The advent of cheaper and more prevalent launch opportunities, combined with the miniaturization of modern electronics, means that putting advanced technology into space is easier today than it has ever been historically. Image sensors now are smaller, cost less, and have higher resolutions than previous generations of space-qualified cameras. These improved devices facilitate cooperative control for clusters of small satellites or space debris detection and avoidance on larger satellites.

In order to further the capabilities of current visual navigation techniques, visible-spectrum images can be used in combination with infrared images. This adds eclipse to the usable portion of the orbit while also extending the operating range to a few hundred meters.

Another way to expand the utility of spacecraft imaging sensors for navigation is to apply terrestrial image processing and recognition techniques. Thus far, these strategies have been beyond the computing capabilities of many smaller budget missions. The appearance matching technique is a powerful algorithm that works particularly well with the on-orbit space environment. It was introduced by Murase and Nayar [1]. The motivation behind the research was to create a technique whereby the recognition system would learn to identify an object from its appearance rather than from a geometric model. Appearance matching has since been used for facial identification [2] and 3D model searches [3].

The appearance matching algorithm also can be extended to the infrared spectrum by adding temperature as an additional parameter. This research applies the technique to spacecraft imaging in both the visible and infrared spectra for the purposes of object identification and relative navigation. Additionally, a simulation environment was created and used to generate the training images necessary for the training step of the appearance matching algorithm.

BACKGROUND

Current technology and algorithms exist that can be leveraged for use in sensor fusion optical navigation. High resolution and high accuracy visual navigation sensors have been flown on numerous satellites, including Orbital Express [4], MESSENGER [5] and HST SM4 [6]. Infrared (IR) sensors have also traditionally been employed for vehicle navigation, for example on the Dragon capsules flown by SpaceX [7]

Combined vision and IR sensors are commonly employed in ground-based security applications [8, 9, 10] and the robustness this field requires can be leveraged for implementation in space. The primary reason for combining images from the two spectra is to extract greater detail in poor or inconsistent lighting conditions. Since the human body is typically much warmer than the surrounding atmosphere, the infrared sensor can be used to determine regions of interest in the image. Those regions can then be applied to the visible-light image for background removal and processing. Alternatively, the infrared image can be used on its own if the lighting in the scene is poor or non-existent.

The nature of infrared and visible light images is such that the former are better for edge detection and the latter for feature detection. These two sensors can be combined into a single image or processed separately in a filter. The ground-based software state-of-the-art includes numerous algorithms for image processing, object identification and recognition, and pose estimation. This existing research can be adapted to the particular challenge of proximity operations in space using both vision and infrared images.

One of the necessary steps to visual navigation is to determine what part of image is the object of interest and what part can be ignored. These regions are referred to as the foreground and background, respectively. For terrestrial applications, this can be a challenging task, especially in the visible spectrum because there are more varied potential backgrounds in the scene. In space, however, the vast majority of on-orbit images will have a black or nearly black background, and a background which will also be much cooler than the object being imaged.

Important to any visual navigation scheme is to define an intended range of operation. If the observer is too close, and the entire object cannot fit within the field of view, recognition is much more difficult. Likewise, when the object gets too far away from the observer, it becomes just a small group of pixels and indistinguishable from stars or other picture artifacts. The ideal range will vary based on the focal length and sensor size of the cameras as well as the size of the target, but generally the operational range in this application will extend from a few meters to approximately 100 meters for visible light and up to 200 meters for infrared without an installed moving lens. Under the assumption of this operating range, distant objects, such as stars or planets, can be easily filtered out given their relative size. The appearance matching algorithm is also robust to shadows giving the appearance of more than one object.

The end result of the image processing is a foreground object presented on a black background. For this reason, the presence of an object is assumed if this filtered image contains any foreground pixels. The emphasis of the current research is therefore

placed on the object recognition and pose estimation portion of the problem. The appearance matching process, which to this point has only been applied to terrestrial, visible images, can be leveraged for infrared measurements and spacecraft relative navigation.

OBJECT RECOGNITION AND POSE ESTIMATION

The use of images for relative navigation consists of answering three questions: is an object present, if so which object is it, and what is its position and orientation with respect to the camera? These questions translate to the tasks of object detection, object recognition, and pose estimation. As previously discussed, when used in space and given a specified operating range, object identification can be a fairly straightforward process.

Single pixels and small pixel groups are assumed not to be part of the foreground when in the proscribed operating range. These artifacts are removed using erosion and dilation. When an image is eroded, any pixel that is the neighborhood of the current background is set to the background. Dilation is the reverse, setting any pixel near current foreground pixel to foreground. When an image is first eroded and then dilated, it is said to be "opened," a process which removes small foreground objects [11]. The additional information provided by the infrared sensor also helps determine the background, since it is generally cooler than the foreground, especially in space applications. These two techniques can be combined for quick and robust foreground determination, as shown in Conaire, et al. [12].
The next step is then to determine which object is present, if the identity is not already established. Previous versions of this research conducted by the authors have used the blobber algorithm [13]. Its main advantages are simplicity and robustness, which come at the expense of accuracy. Part of that simplicity is the fact that the algorithm cannot perform object recognition. The geometry of the target object must be provided beforehand.

Appearance matching [1] is an approach to object recognition and pose estimation that analyzes the image of the object pixel by pixel as whole, as opposed to identifying points of interest. Training images are vectorized and mapped as points in a parametric eigenspace, where the parameters are the object pose and illumination. These points can then be interpolated into a hypersurface in the eigenspace. A new images is mapped into the same eigenspace, and its location and proximity to the hypersurface is used to determine the identity of the object in the image.

The final step is pose estimation. The blobber algorithm approaches the relative navigation problem by analyzing the geometric properties of the image foreground, which is subject to error when the body is only partially illuminated. This error results from the fact that while the blobber algorithm takes into account varying orientations of the object, all of the orientations are viewed with the same illumination conditions.

Appearance matching performs pose estimation by mapping the test image into an eigenspace specific to the object which was identified previously. Its location and proximity to the object-specific hypersurface determines the orientation and lighting of the object in the test image. This process takes into account the relative brightness of the pixels and thus is robust to varying lighting conditions, as long as they are similar to the training images. The appearance matching algorithm's robustness to varying illumination is the primary reason it was selected. Harsh or inconsistent illumination of the object can be accounted for by using simulated or actual training images under the difficult conditions.

Algorithm overview

First, the training image set is normalized. This process begins by cropping the image to include as little of the background as possible while maintaining a square aspect ratio. This result is then scaled to a specified size, for example 100 by 100 pixels. Finally, these images are vectorized by reading the brightness values in a raster scan pattern to create an image vector. In order to account for varying lighting intensity, each image vector is scaled so that the total energy in the image is unity, resulting in the normalized image vector $\boldsymbol{x}$

$$\boldsymbol{x} = [x_1, x_2 \dots, x_n]^T$$

The normalized image vectors are combined side by side to form image sets. The universal image set consists of every training image, while the object sets contain all of images of a single object. Similarly, the average image $\boldsymbol{c}$ can be calculated by found by summing each column of the image set and dividing by the total number of images, while the object average image $\boldsymbol{c}^{(p)}$ is found by averaging the image vectors of each object.

These averages are used to construct both the universal eigenspace, which is calculated from every image in the training set, and the object eigenspaces, which are calculated using only images from each object. The image matrix $\boldsymbol{X}$ is found by subtracting the average image from each $\boldsymbol{x}$ and combining them as shown in Equation [ref]. This leads to the covariance matrix $\boldsymbol{Q} \triangleq \boldsymbol{X}\boldsymbol{X}^T$.

$$\boldsymbol{X} \triangleq \left\{\boldsymbol{x}_{1,1}^{(1)} - \boldsymbol{c}, \dots, \boldsymbol{x}_{R,1}^{(1)} - \boldsymbol{c}, \dots, \boldsymbol{x}_{R,L}^{(P)}\right\}$$

The eigenvalue problem is solved for $\boldsymbol{Q}$. Finding the full set of eigenvalues would be impractical, so only the first k pairs of eigenvalues λ_i and eigenvectors $\boldsymbol{e}_i$ are calculated. According to Murase and Nayar [1], a value for k of 10 or fewer is sufficient for learning and recognition. Finally, for each image in the universal set, the point $\boldsymbol{g}_{r,l}^{(p)}$ is located as a point in the k-dimensional eigenspace (Eq. [eq:grlp]). These points are interpolated from the hypersurface $\boldsymbol{g}(\theta_1, \theta_2)$, where θ_1 and θ_2 are the indices of the orientation and illumination conditions, respectively.

$$\boldsymbol{g}_{r,l}^{(p)} = [\boldsymbol{e}_1, \boldsymbol{e}_2, \dots, \boldsymbol{e}_k](\boldsymbol{x}_{r,l}^{(p)} - \boldsymbol{c})$$

The images of each object p can be projected into its object eigenspace by creating the object specific covariance matrix $\boldsymbol{X}^{(p)}$ and calculating the resulting eigenspace from the object covariance matrix $\boldsymbol{Q}^{(p)} \triangleq \boldsymbol{X}^{(p)}(\boldsymbol{X}^{(p)})^T$.

$$\boldsymbol{X}^{(p)} \triangleq \left\{\boldsymbol{x}_{1,1}^{(p)} - \boldsymbol{c}, \dots, \boldsymbol{x}_{R,1}^{(p)} - \boldsymbol{c}\right\}$$

The first k pairs of eigenvalues $\lambda_i^{(p)}$ and eigenvectors $\boldsymbol{e}_i^{(p)}$ are calculated as well the hypersurface $\boldsymbol{f}^{(p)}(\theta_1, \theta_2)$ from the points $\boldsymbol{f}_{r,l}^{(p)}$ (Eq. [eq:frlp])

$$\boldsymbol{f}_{r,l}^{(p)} = \left[\boldsymbol{e}_1^{(p)}, \boldsymbol{e}_2^{(p)}, \dots, \boldsymbol{e}_k^{(p)}\right](\boldsymbol{x}_{r,l}^{(p)} - \boldsymbol{c}^{(p)})$$

The preceding steps are part of the initialization of the algorithm. They are performed only once and the results are stored. For each new image received, the algorithm maps it to the universal eigenspace for object recognition. First, the location $\boldsymbol{z}$ of the new image in the universal eigenspace is found. The object p that minimizes the distance between $\boldsymbol{z}$ and the universal hypersurface $\boldsymbol{g}(\theta_1, \theta_2)$, is determined.

$$\boldsymbol{z} = [\boldsymbol{e}_1, \boldsymbol{e}_2, \dots, \boldsymbol{e}_k]^T(\boldsymbol{y} - \boldsymbol{c})$$
$$d_1^{(p)} = \min_{\theta_1, \theta_2} ||\boldsymbol{z} - \boldsymbol{g}(\theta_1, \theta_2)||$$

If $d_1^{(p)}$ is within a specified threshold, then the image is concluded to be of object p. In order to perform pose estimation, the image is then mapped into eigenspace of object p. The indices θ_1 and θ_2 that result in the minimum value of $d_1^{(p)}$ define the pose of the object.

$$\boldsymbol{z}^{(p)} = \left[\boldsymbol{e}_1^{(p)}, \boldsymbol{e}_2^{(p)}, \dots, \boldsymbol{e}_k^{(p)}\right]^T (\boldsymbol{y} - \boldsymbol{c}^{(p)})$$
$$d_2^{(p)} = \min_{\theta_1, \theta_2} ||\boldsymbol{z}^{(p)} - \boldsymbol{f}^{(p)}(\theta_1, \theta_2)||$$

Advantages of appearance matching

The appearance matching algorithm [1] accounts for the main issues with the blobber algorithm described previously. The learning step involves the measurement or simulation of the target object at various orientations and illuminations. These parameters are only varied slightly from image to image. Images that have similar orientation or illumination conditions are highly correlated and therefore map to nearby locations in a multi-dimensional eigenspace. Features on the object lead to distinctive shading and shadowing on the body that is used to locate a new image in the space. It also mitigates the problem with partial illumination and variable lighting conditions, since the number, location, and intensity of foreground pixels is part of the "signature" of that pose and illumination state. Figure 1 shows a simulated image of a cube and its corresponding lighting signature. Finally, the additional object recognition step means the algorithm can recognize the object in the image from a library of multiple objects instead of being limited to just one option.

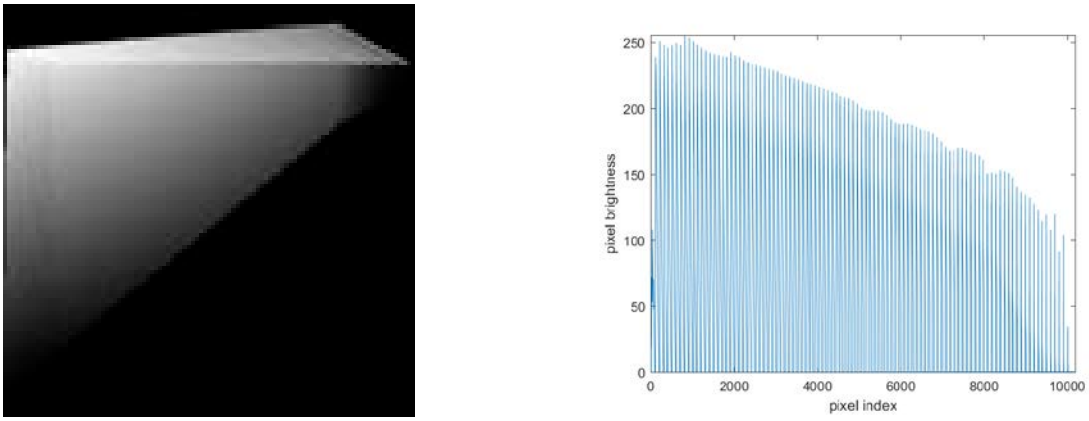


Figure 1 – Simulated image (a) and brightness signature (b)

Application to infrared

Another advantage to employing appearance matching is that it can be extended to infrared images. In the training step, images are generated at different orientations and illuminations. To apply this to the infrared spectrum, it is only necessary to add temperature as an additional dimension to this training image space. Instead of being a function of two indices, the universal hypersurface and the object hypersurfaces are now functions of three indices. There is an additional computational cost to generating another dimension of training images. However, if the thermal behavior of the object's material is well understood, it is possible that only a few temperature values need to be generated; the rest can be interpolated. This topic will be more fully explained with algorithms in the final version of the paper.

SPACECRAFT IMAGING SIMULATION ENVIRONEMENT

Simulation is a key aspect to the implementation of the appearance learning object identification procedure. In an ideal scenario, the images necessary for the learning step of the algorithm would come from actual images of the object using the camera and bolometer. However, the possibility exists that such images are unavailable before launch. The desired target could have been launched after the visual navigation system is on orbit, or it could be an object like space debris that varies in size and shape. Obtaining real images is also time and labor intensive. Therefore, a modeling environment which can create realistic images under various orientations and lighting conditions is necessary for the full implementation of the appearance matching algorithm.

In addition to providing training data, the simulation environment can also be used to perform software-in-the-loop tests to verify the accuracy of the object recognition and the resulting relative navigation solution. In order to provide images for these two applications, a MATLAB script was written called the spacecraft imaging simulation environment (SISE). The purpose of the SISE is to generate a series of simulated images of a target spacecraft as viewed by a sensing spacecraft. A preliminary version was developed for the work performed by McBryde and Lightsey [5]. The software tool has since been

refined to improve the speed of simulated image generation as well as to incorporate various errors sources which may be present in actual images. A block diagram is shown in Figure 2.

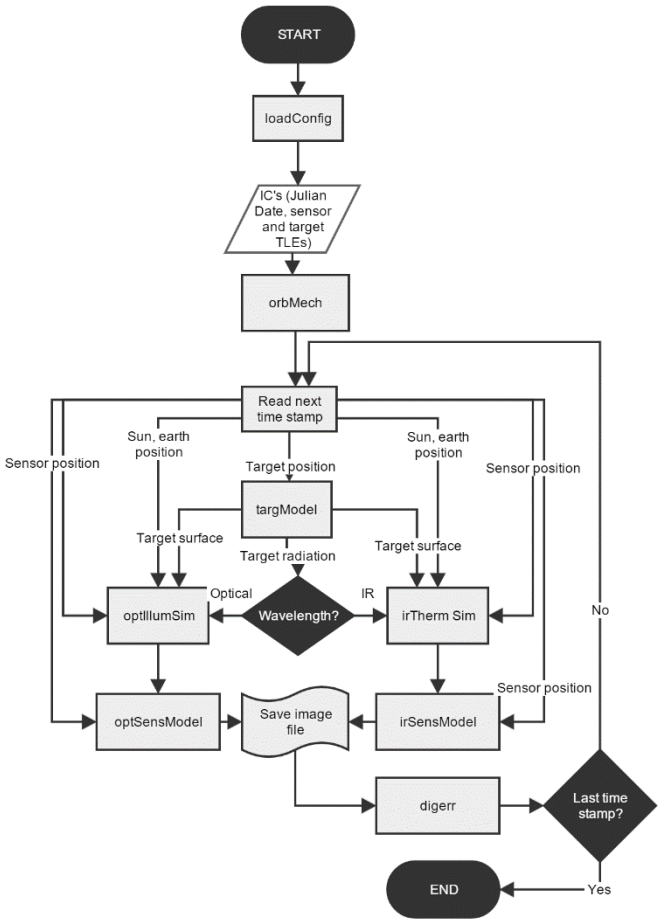


Figure 2 - SISE block diagram

Target Modeling

The target modeling subroutine, *targModel* populates a target structure from given parameters. These parameters include the size and shape of all of the primitive components of the object, their optical and thermal properties, and the starting temperature for the object. The resulting *targ* structure includes the location in space of each vertex of the object, the optical and thermal attributes of each vertex, and the list of vertices that make up each object face. Note that this structure is solely a function of the object properties, location, and orientation, and thus do not have to be recalculated if only the position of the camera is changing.

Radiation simulation

The radiation simulation subroutines predict the visible or infrared light emanating from each vertex on the target object. This radiation is function of the location, normal direction and optical properties of the vertex; the relative location of the illumination sources in the scene; and, for infrared radiation, the thermal properties and temperature of the object. For the types of radiance that are dependent on the direction to the observer (i.e., specular reflection), that vector is taken into account as well. Attributes are then added to the *targ* structure for visible spectrum radiance. Finally, the temperature of each vertex is updated based on energy absorbed, emitted, and conducted to neighboring vertices. There are two subroutines, *optIllumSim* and *irThermSim*, which simulate the visible and infrared spectra, respectively. Equation [eq:radiation] gives a

general framework for the calculation of spectral radiance from a vertex. It was developed based on equations from Christian [6] and Garnier, et al. [7].

$$L_{\boldsymbol{P}_o}(\lambda) = k_a L_a(\lambda) + \sum_{l=1}^{n} L_l(\lambda)(k_d(N \cdot I) + k_s(E \cdot V)^{\alpha}) + \delta_e L_t(\lambda)$$

Sensor measurement

The last required step is to calculate the radiation incident on the sensors and produce images. The sensor model subroutines go pixel by pixel and calculate if objects are seen by the pixel, which of the observed objects is the closest, and how much the object irradiates that pixel. Then, the subroutines translate that irradiance into a greyscale brightness values based on the internal physics of the sensors. The final result is a simulated image which can they be used either for appearance matching learning or a software-in-the-loop simulation. Equation [eq:irradiance] gives the general framework for irradiance on a pixel, while Equation [eq:measurement] relates that to a greyscale value in the final image. These two equations were also based on work from Christian [6] and Garnier, et al. [7].

$$S_{\boldsymbol{P}_i}(\lambda) = L_{\boldsymbol{P}_o}(\lambda)\tau(\lambda)\frac{A_{ap}}{f^2}\cos^4\theta$$

$$s = \int_T \int_{\lambda_{\min}}^{\lambda_{max}} A_{det} S_{\boldsymbol{P}_i}(\lambda) FR(\lambda) d\lambda$$

The sensor measurement subroutine is also where the first of the error sources in implemented: radial distortion. Since this effect bends light that enters the outside of the lens, a distortion function shifts the incident radiation from each vertex toward the outside of the image, based on the level of radial distortion required.

Errors

After the image has been simulated, the other possible error sources can be added, if needed. One potential error source is pixel blur, either due to defects in the lens or object motion. This effect can be simulated by a convolution function based on the amount of required blur, which is then iterated over each pixel in the image. Shot noise can also be introduced using a point spread function based on a normal distribution. Finally, amplifier and digitization noise can be added. These effects are functions of the dynamic range and the saturation limit, respectively.

Verification

The Ames Research Center optics lab was used to verify the SISE. Figure 3 shows an actual image taken of a satellite analog and its simulated counterpart. The analog was a 3-inch aluminum sphere at a distance of approximately 44 cm. Verification tests were performed and these tests show a strong correspondence between the simulated and actual images (Figure 4). The average pixel error was 1.24 out of 255.

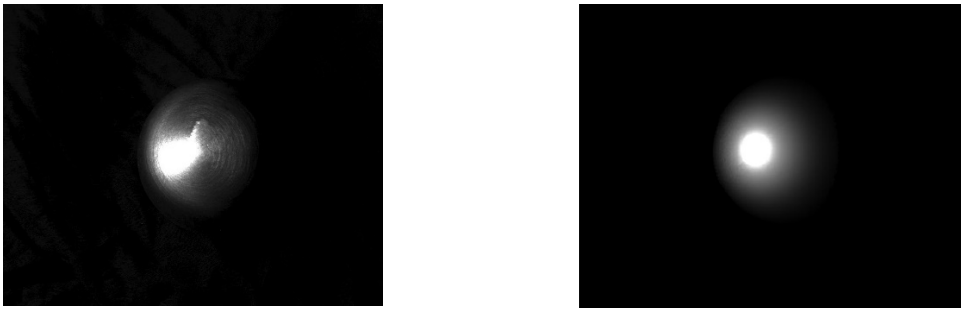


Figure 3 - Actual image (a) and simulated image (b)

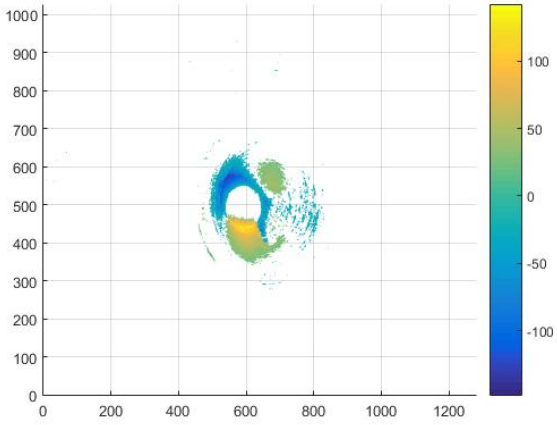
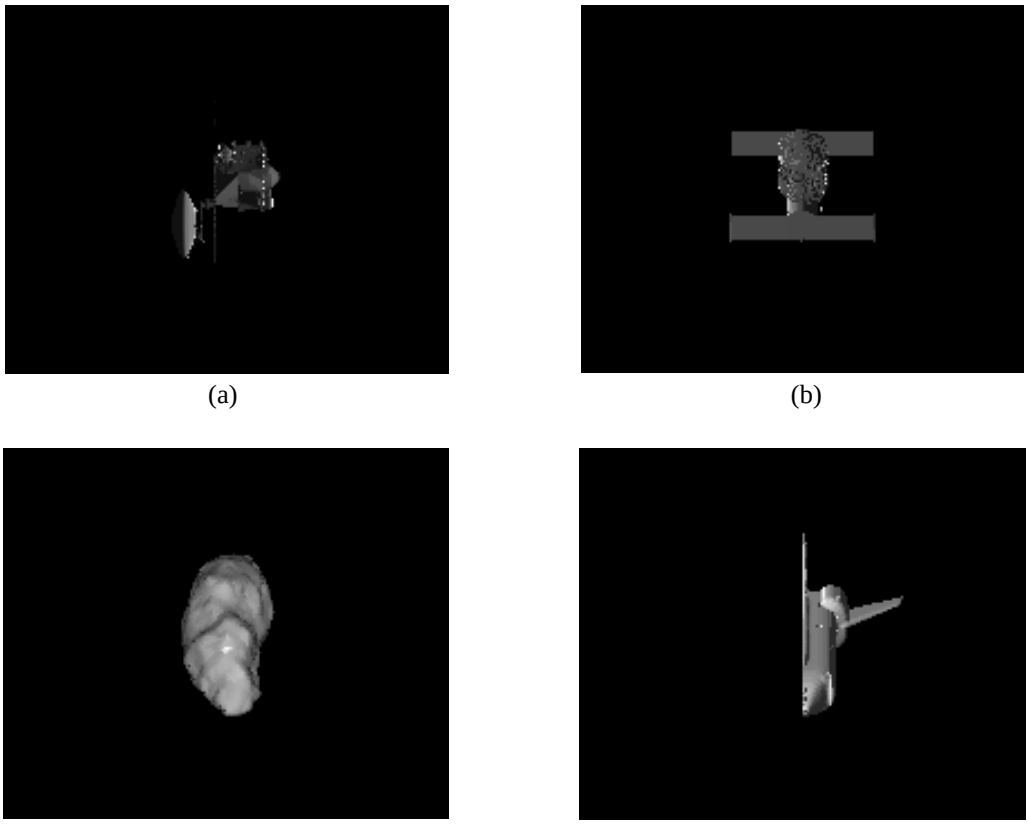


Figure 4 - Verification results

RESULTS

In order to test the accuracy of the appearance matching algorithm, four objects of different shapes were simulated in various poses and illumination conditions. In order to represent potential on-orbit tasks for this visual navigation system, these objects were selected to be the Mars Reconnaissance Orbiter (MRO), the Hubble Space Telescope (HST), the asteroid Geographos, and the Space Shuttle orbiter. An example training image of each is shown in.



(a) (b)

(c) (d)

Figure 5 - Simulated images of MRO (a), HST (b), Geographos (c), and Space Shuttle orbiter (d)

Training data sets of each of these models were generated using the SISE under different conditions. These data sets were structured so as to test the effect of various parameters on the object identification and pose estimation performance: number of orientations, number of lighting conditions, and spectrum (visible, infrared, or fused). The results of the object recognition tests are given in Table 1, Table 2, and Table 3. Pose estimation results are shown in Table 4, Table 5, and Table 6.

Object Recognition

Table 1 - Visible spectrum recognition accuracy percentage

	4 light	8 light
20 orientations	74	97
40 orientations	74	98
60 orientations	74	100

Table 2 - Infrared spectrum recognition accuracy percentage

	4 light	8 light
20 orientations	100	100
40 orientations	100	100
60 orientations	100	100

Table 3 – Fusion recognition accuracy percentage

	4 light	8 light
20 orientations	96	100
40 orientations	96	100
60 orientations	96	100

The first thing to note is how strong the performance of the infrared images is in object recognition. Despite changing the number of training orientations and lighting conditions, the infrared object recognition is perfect. It does logically follow that the lighting would not have a large effect on the infrared, since most of the radiation is coming from the heat of the body itself. Still, infrared is clearly the best way to identify an object if that signal is available. It should also be observed that number of illumination conditions trained had a much greater effect on the accuracy of the visible spectrum recognition than did number of training orientations.

Pose Estimation

Table 4 – Average visible spectrum pose error (degrees)

	4 light	8 light
20 orientations	31.027	24.584
40 orientations	3.6322	3.2949
60 orientations	4.0315	3.2868

Table 5 – Average infrared spectrum pose error (degrees)

	4 light	8 light
20 orientations	31.027	26.014
40 orientations	3.3881	1.5345
60 orientations	4.0315	1.5435

Table 6 – Average fusion pose error (degrees)

	4 light	8 light
20 orientations	32.302	25.647
40 orientations	3.3180	1.5362
60 orientations	3.3828	1.4527

The pose estimation tests yielded some interesting results. First, there is a clear threshold of performance when it comes to the number of training orientations necessary for good pose estimation. Every case shows a significant drop-off between 20 and 40 training orientations. As one might expect, the number of lighting conditions did not have as much of an effect as number of orientations when it came to pose estimation performance. However, the cases with 8 lighting conditions displayed approximately half of the error as the 4 lighting condition cases for the 40 and 60 orientation cases. As above, we can see an improvement in nearly every case with the application of sensor fusion, though at times the gains are marginal.

One surprising result was the better average performance of the infrared images versus the visible spectrum images. Given the greater detail in the visible images, they should perform better for pose estimation. Upon analyzing the results, it became apparent that the third object, the asteroid, had returned some pose errors near plus or minus pi. These large errors are clearly the result the symmetricity of the asteroid. Once those results are removed, the visible spectrum images have similar performance to those in the infrared spectrum.

Finally, it is worthwhile to compare the performance between types of object. The MRO, HST, Geographos, and orbiter had the following average performance for the 60 orientation and 8 lighting condition case (Table 7)

Table 7 - Average pose error for 60 and 8 case (degrees)

MRO	HST	Geographos	Space Shuttle Orbiter
1.339567	1.6348	3.908467	1.494533

The object's performance, as expected, is arranged in rough order of symmetricity, with Geographos having the worst performance and MRO the best.

CONCLUSION

The SISE will be a useful tool for software-in-the-loop verification, tuning of the relative navigation algorithm, and generating training images. Application of sensor fusion extends the usability of visual navigation into the eclipse portion of the orbit as well as augmenting the accuracy of object identification and pose estimation. By employing appearance matching, this system can provide training data for objects that cannot be imaged on the ground and return accurate relative pose information.

In terms of future work, the actual fusion filter needs to be improved and refined. In theory, this kind of filter can mitigate the effects of symmetricity of the object. Also, the system should be tested with real world images. A follow-up test using 3-D printed models is planned.

ACKNOWLEDGEMENTS

This work was sponsored in part by NASA contracts NNX09M51A and NNX15AD26H.

REFERENCES

[1] H. Murase and S. K. Nayar, "Learning and recognition of 3D objects from appearance," in *Proceedings of IEEE Workshop on Qualitative Vision*, 1993.

[2] X. He, S. Yan, Y. Hu, P. Niyogi and H. J. Zhang, "Face recognition using Laplacianfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 328-340, March 2005.

[3] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin and D. Jacobs, "A search engine for 3D models," *ACM Transactions on Graphics,* vol. 22, no. 1, pp. 83-105, January 2003.

[4] R. T. Howard, A. F. Heaton, R. M. Pinson and C. K. Carrington, "Orbital Express Advanced Video Guidance Sensor," in *IEEE Aerospace Conference*, 2008.

[5] K. Williams, A. Taylor, B. Page, P. Wolff, B. Williams, D. Stanbridge and J. McAdams, "Navigation for the MESSENGER Mission's First Mercury Encounter," in *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, American Institute of Aeronautics and Astronautics, 2008.

[6] B. Naasz, J. V. Eepoel, S. Queen, C. M. Southward and J. Hannah, "Flight Results of the HST SM4 Relative Navigation Sensor System," in *33rd ANNUAL AAS GUIDANCE AND CONTROL CONFERENCE*, 2010.

[7] *Space Exploration Technologies Corporation - Dragon.*

[8] J. Han and B. Bhanu, "Fusion of color and infrared video for moving human dectection," *Pattern Recognition,* vol. 40, no. 6, pp. 1771-1784, June 2007.

[9] J. Saeedi and K. Faez, "Infrared and visible image fusion using fuzzy logic and population-based optimization," *Applied Soft*, vol. 12, no. 3, pp. 1041-1054, 2012.

[10] J. Zhao and S. S. Cheung, "Human segmentation by geomgeometric fusing visible-light and thermal imageries," *Multimedia Tools and Applications,* vol. 73, no. 1, pp. 61-89, 2014.

[11] J. C. Christian, "Optical Navigation for a Spacecraft in a Planetary System," 2011.

[12] C. Ó. Conaire, E. Cooke, N. O'Connor and N. Murphy, "Background Modelling in Infrared and Visible Spectrum Video for People Tracking," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005.

[13] L. Walker and D. Spencer, "Automated proximity operations using image-based relative navigation," in *26th Annual USU/AIAA Conference on Small Satellites*, 2012.

[14] C. R. McBryde and E. G. Lightsey, "End-to-End Testing of a Dual Use Imaging Sensor for Small Satellites," *Journal of Small Satellites,* vol. 5, no. 1, pp. 435-448, Feburary 2016.

[15] C. Garnier, R. Collorec, J. Flifla, C. Mouclier and F. Rousée, "Infrared Sensor Modeling for Realistic Thermal Image Synthesis," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1999.