AUTONOMOUS ON-BOARD NEAR EARTH OBJECT DETECTION

P. Rajan[1] *P. Burlina*[1,2] *M. Chen*[2] *D. Edell*[2] *B. Jedynak*[3] *N. Mehta*[2] *A. Sinha*[4] *G. Hager*[1]

The Johns Hopkins University [1]Dept. of Computer Science, [2]Applied Physics Laboratory
[3]Dept. of Applied Mathematics and Statistics [4]Dept. of Electrical and Computer Engineering

ABSTRACT

Most large asteroid population discovery has been accomplished to date by Earth-based telescopes. It is speculated that most of the smaller Near Earth Objects (NEOs) up to 100 meters in diameter, whose impact can create substantial city-size damage, have not yet been discovered. Many asteroids cannot be detected with an Earth-based telescope given their size or the location of the Sun. We are investigating asteroid detection algorithms that can be hosted on-board a spacecraft, thereby minimizing the expense and need to downlink entire images. Having autonomous on-board algorithms enables the ability to deploy a spacecraft at approximately 0.7 AU heliocentric or Earth-Sun L1/L2 halo orbits, removing some of the challenges associated with detecting asteroids with Earth-based telescopes. We describe an image processing algorithm developed for on-board asteroid detection and show that its performance is consistent with deployment on flight-qualified hardware.

***Index Terms*—** Asteroids detection, identification.

1. INTRODUCTION

NASA has a congressional mandate to discover all Near-Earth Objects (NEOs) at least 1 kilometer in diameter. Fortunately, 95% of the NEOs larger than 1 km that have been discovered are likely not to impact Earth. Near-Earth Object search programs [1] are currently almost exclusively accomplished by Earth-based telescopes such as MIT's LINEAR [2] project, the NEAT [3] program, the Catalina Sky Survey [4], or Pan-STARRS [5]. An exception is JPL's spacecraft-based WISE telescope brought out of hibernation to characterize NEOs in the 3.4 and 4.6 micron infrared bands [6] (called NEOWISE).

It is notable, however, that the NEO that impacted Chelyabinsk, Russia on 15 February 2013 was only about 17 meters. It is estimated that only a relatively small fraction of those so called "city-killing" asteroids, particularly objects less than 100 meters in diameter, have been discovered to date [7]. Because of their size, atmospheric effects and the location of the sun, some of these NEOs cannot easily be detected with an Earth-based telescope.

Our focus here is therefore on developing an algorithm that can be hosted on-board a spacecraft. The two nominal approaches that we are investigating include deploying the algorithm on a BAE RAD750 with 8 MB of EEPROM and 32 MB RAM in addition to Virtex 5 FPGAs. Understanding that there are processing and memory constraints on-board, our algorithm design needs to not only meet the performance objectives of detecting and identifying asteroids, but its implementation needs to be optimized for the respective platforms. This paper describes an agile algorithm candidate that is being investigated as well as our use of representative real and simulation imagery for testing it.

Notable prior work includes [5], describing a reference processing system for detection and identification of asteroids named the Pan-STARRS Moving Object Processing System (MOPS). This pipeline identifies moving objects in our solar system and links those detections within and between night observations. It attributes detections to known objects, calculates initial and differentially corrected orbits for linked detections, recovering detections when they exist, and orbit identification. Most proposed pipelines for Earth-based detection include a step to combine images into a high S/N static-sky image that is subtracted from the current master image to obtain a difference image containing only transient sources. Examples include [8], where a shift-and-add technique is used to improve signal to noise ratio and then synthetically creating long exposure images to facilitate the detection of trajectories. A related shift-and-add method using a median image rather than an average image is reported in [9]. A match filter is used for asteroid detection and matching in [10]. In [11, 12, 13], tree based searches (including KD-trees) are used for efficient linking of successive asteroids detections and finding sets of observation points that can be fitted with an inherent motion model, through an exhaustive search for all possible linkages that satisfy the model constraints.

This paper describes a small-footprint pipeline (with regard to memory and CPU usage) that can be deployed on flight-qualified hardware. When compared to prior studies, novel and salient contributions of our work include: the use of Principal Component Analysis (PCA) and 2d-trees to efficiently link detections into trajectories while mitigating false

This work was supported by an Early Stage Innovations grant from NASA's Space Technology Research Grants Program NNX14AB04G. We gratefully acknowledge E. Christensen for providing the CSS data for this study.

positives (Section 2); and the use of a simulation engine relying on principled optics models to characterize performance on realistic space-based imagery, along with a combination of real image datasets (NEAT and CSS) (Section 3).

2. APPROACH

The main components of our image processing pipeline addresses detection and linking of asteroids and takes as input a sequence of time-lapse images acquired from space-based platforms, as described (also detailed in Algorithm 1):

Image Pre-Processing Median filtering is applied to reduce impulsive noise and artifacts that depend on the acquisition.

Image Registration Image registration is used to bring the images into alignment with a common image of reference so that the stars in the background line up in all images. In the case of a triplet of images, the second image is used as reference. We use a similarity transformation (translation, rotation, scaling) and/or skew (full affine transformation) to align all images in the sequence. Our registration uses mutual information [14] to estimate the transformation parameters.

Image Logical Differencing A global thresholding is applied to the registered image for detecting asteroids and suppressing background noise. As asteroids are typically very faint compared to the surrounding stars, the selection of the detection threshold impacts false alarm rate. Thresholding yields binary detection images. The set of all binary images is then used to generate an intersection image that contains objects that occur in at least two images in the sequence. This is followed by logical differencing whereby we produce a set of difference images by intersecting the corresponding binary image with the logical complement of the common intersection image. This operation provides a list of candidate detections (movers) for each image in the sequence. While the logical differencing results in good detections, additional artifacts such as crater-like formations (see Fig. 5) are seen as a result of some celestial bodies being over-exposed. To mitigate this artifact, we find the connected components (using 8-neighborhood connectivity) in each difference image and filter out hollow objects based on connected component size and extent, and store the centroids of the selected components for the next stage of the pipeline.

Trajectory Linking The list of centroids of moving objects obtained from image differencing defines a set of candidate rectilinear trajectories. The goal is to find a subset of centroids that fit a linear model. The set of filtered centroids potentially has a high number of noisy points (falsely detected movers), and the cardinality of the set of all candidate trajectories increases exponentially with the number of detections, thus requiring subsequent pruning. We therefore combine PCA and 2d-trees to efficiently find trajectories. Unlike MOPS [5] and CSS[4], we do not set an upper limit on the velocity of the asteroid, and hence do not risk missing fast movers. Given a sequence of images, we form all the possible trajectories connecting the detections in the first and last frames. There are $O(n^2)$ such trajectories, where n is the average number of detections per image. We then calculate the points of intersection of each of these trajectories with all the frames in the middle. A binary space partitioning 2 dimensional tree is constructed from the candidate detections for each of the middle frames(excluding the first and last frames), and we perform a range search on the trees to find a subset of points that lie near the point of intersection in each frame. Each node of the tree stores the bounding box of all its children in order to make the search efficient. The range search query can be done in $O(\log(n))$ time on average. Once we find a collection of such points that potentially form linear trajectories, we perform PCA and compute the ratio of eigenvalues to develop a line confidence score for each candidate trajectory, and choose lines that have high confidence. Using the candidate trajectories thus found, we then enforce temporal ordering by using the sign of the projection on the principal eigenvector. As a final step, we eliminate false positives by ensuring that the distance between projections is proportional to the time interval between images.

Algorithm 1 Detection and Linking Algorithm for a sequence of images

1: Apply a median filter to each image in the sequence.
2: Align all the images to a reference frame (e.g. the second image for a triplet) using Mutual Information based registration.
3: Perform thresholding on the registered images using a predefined threshold to generate binary detection images.
4: Form the logical intersection image from all the binary detection images, then perform logical differencing between each binary detection image and the common intersection image.
5: Find the connected components in the difference images, filter out objects based on connected component area and extent to produce a set of candidate moving objects in each image.
6: Form the set of all trajectories connecting the candidate moving object detections from the first and last frame.
7: Construct 2d-trees for the difference images corresponding to the frames in the middle, excluding the first and last frames.
8: **For all** trajectories connecting the first and last frames **do**
   - Find the points of intersection of each trajectory with the middle frames.
   - Perform range search on the 2d-trees to find a list of detections near this point of intersection within each difference image.
   - Perform PCA on this subset of trajectories connecting points from all frames in the sequence. Find the ratio of eigenvalues, $\lambda_1/(\lambda_1+\lambda_2)$. Reject trajectories for which this ratio is below a line confidence threshold. Ensure that the distance between projections is proportional to the time interval between images.

3. EXPERIMENTS

We detail the experiments performed using a range of simulated space-based imagery generated using the JHU/APL de-

Fig. 1. Left: An image simulated by RCE. Right: 31 simulated MWIR images super-imposed in order to visualize the trajectory of the asteroid in a single image. The true trajectory can be seen as a faint line towards the top center of the image.

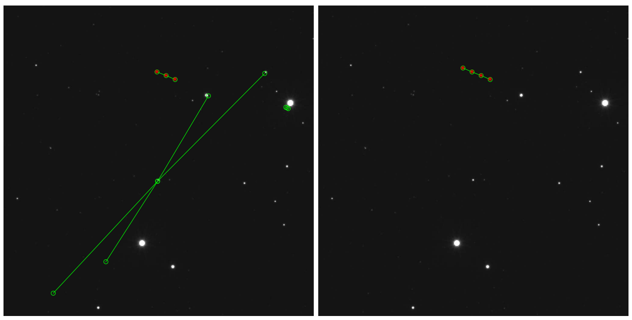

Fig. 2. The trajectories found by the pipeline are shown in green. The true location of the asteroid is marked in red.
Left: Trajectory Detection on a simulated triplet. Right: Trajectory Detection on a quadruplet. Adding one more image to the triplet eliminates the false positives.

veloped *Renderer and Camera Emulator* (RCE) as well as real imagery from the NEAT and Catalina Sky Survey dataset.

3.1. Simulated Space-Based Imagery

Using RCE, asteroids are modeled as spherical blackbody-like emitters, with a cross-sectional area that approximates the sizes of actual asteroids and surface temperatures typical of sun-illuminated asteroids in an Earth-like orbit. Similar to the way stars are modeled, the radiation emitted is modeled using a form of Planck's equation:

$$B_\lambda(T) = \epsilon \frac{2hc^2}{\lambda^5} \frac{1}{e^{\frac{hc}{\lambda k_B T}} - 1} \tag{1}$$

where $B_\lambda(T)$ is the spectral radiance at a given wavelength λ and temperature T The value ϵ is the emissivity of the asteroid, which essentially converts the blackbody spectral radiance into spectral irradiance. The constant, h is the Planck's constant, c is the speed of light, λ is wavelength, k_B is the Boltzmann constant, and T is the temperature.

The asteroids are assumed to have a nominal temperature of 200 K due to solar heating and emissivities in the range from 0.9 to 0.98. The RCE uses stellar data available as part of the Two Micron All Sky Survey (2MASS), a stellar survey that scanned the entire sky in three IR bands (centered at 1.25 µm, 1.65 µm, and 2.17 µm, respectively). The 2MASS catalog also incorporates data in two visible bands from other surveys. An example of the simulated MWIR image and the ground truth trajectory derived from one set of simulated imagery is shown in Fig. 1. Fig. 2 shows the final trajectories detected by our algorithm for one triplet and one quadruplet of the simulated MWIR dataset. As is shown in Fig. 2, using trajectory verification on a greater number of images in the sequence allows us to quickly disambiguate and reject false trajectories. In this case this trend is readily apparent when going from a triplet to a quadruplet of images.

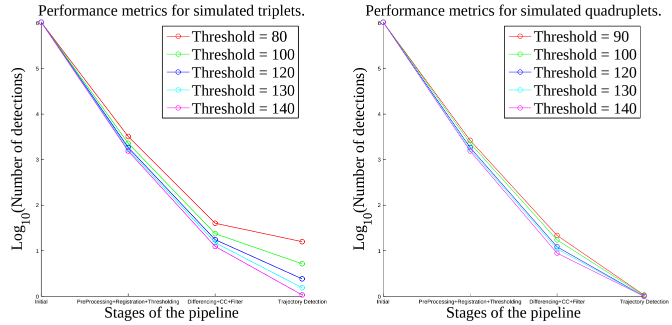

Fig. 3. Left: The number of detections at various stages of the pipeline for triplets of images. Right: The number of detections at various stages of the pipeline for quadruplets of images.

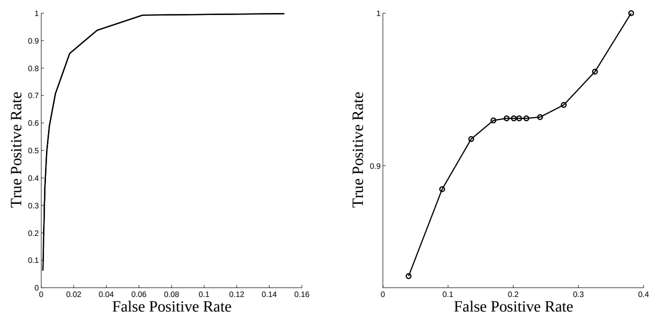

Fig. 4. Left: ROC curve for the detection of stars and asteroids after the image thresholding stage of the pipeline. Right: ROC curve for asteroid detections at the final stage of the pipeline.

When testing on a set of 31 triplets or quadruplets taken from simulated space-based imagery, we characterize the algorithm with regard to detections at each of the following three successive stages: the initial detection of objects, the detection of moving objects and the detection of trajectories. In Fig. 3, we plot the number of detected objects at each step of the pipeline for each selected threshold value. We note in the right plot in Fig. 3 that the use of quadruplets allows for a single trajectory to be found irrespective of the number of detections found at prior stages echoing the results displayed in Fig. 2, and showing that false positive rate can be completely abated in this case. Last, we show the Receiver Operating Characteristic (ROC) obtained for triplets in Fig. 4. Since the use of quadruplets yields no false alarms we do not show the corresponding ROC curve.

3.2. Real Imagery

3.2.1. NEAT

Near Earth Asteroid Tracking (NEAT) [3] is an earth-based program run by NASA from 1995-2007 to discover NEOs. Fig. 5 and Fig. 6 show the results at all stages of the pipeline for one triplet of images of the 2002-CY46 asteroid obtained from the NEAT system archive.

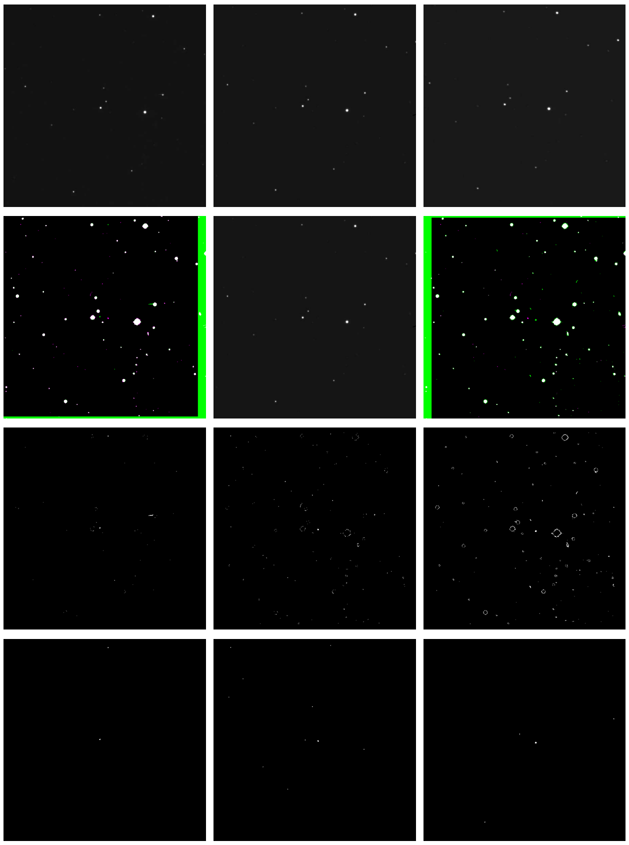

Fig. 5. Image Processing Pipeline results shown on 2002 NEAT data. (row 1): input image triplet (CY46) taken approx. 10 minutes apart. (row 2): Image Registration. Left: Image-1 registered to Image-2. Right: Image-3 registered to Image-2. (row 3): Image Differencing: Artifacts such as crater-like formations are seen in the difference images above. This is the result of some celestial bodies being overexposed.) (row 4): Image Differencing: Filtered centroids shown in each image of the sequence.)

3.2.2. CATALINA

The Catalina Sky Survey (CSS) [15, 16] is intended to discover NEOs, specifically potentially hazardous asteroids that pose risk to earth. Fig. 7 shows the final result for one set of earth-based images from CSS.

3.3. Implementation

Algorithmic development and results shown earlier were performed in MATLAB. The pipeline is being jointly im-

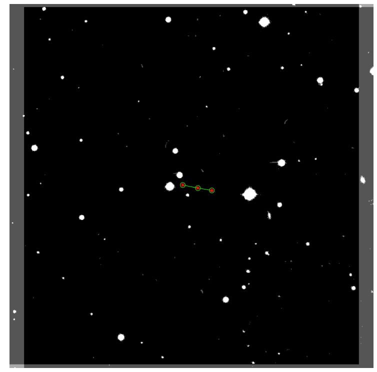

Fig. 6. Trajectory Detection for the NEAT CY46 Triplet. Asteroid trajectory detected is shown in green. True location is in red. 3 Images of the triplet are super-imposed here after registration and thresholding for ease of visualization.

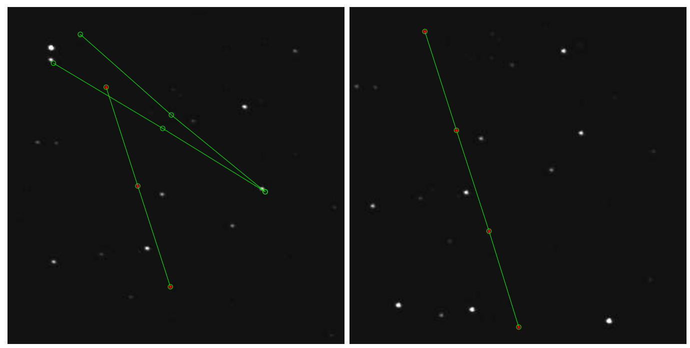

Fig. 7. Results shown on CSS data. The registered images are super-imposed to visualize the trajectory in a single image.Asteroid trajectory detected is shown in green, true location of the asteroid is in red. Left: Trajectories detected on a triplet. Right: Trajectory detected on a quadruplet.

plemented in C++ for benchmarking on a MCP750, a good proxy for the RAD750. The MCP750, with clock speed of 367 MHz, is completely dedicated to the algorithm and not servicing spacecraft or instrument commands during benchmarking. Run time for steps 2-8 (minus the 2d-trees) on a quadruplet is about one second. Preliminary tests indicate that 16 MB of RAM is used and the current executable is 1.3 MB. Any preprocessing such as step 1 would be implemented on an FPGA. While further optimization is under way, current results indicate that this algorithm can be deployed on a spacecraft processor.

4. CONCLUSION

We developed a pipeline for space-based asteroid detection. Its performance and run time make it a good candidate for deployment on an on-board processor. We also demonstrated that it is able to abate false positives by using a combination of PCA and 2d-trees on quadruplets. Current run time also supports a mission concept that aims for image acquisition at a 10 minute cadence. Providing a capability to autonomously detect asteroids on a space-based processor opens up the possibility of deploying a space-based telescope to a Venus-like orbit for NEO Surveys.

5. REFERENCES

[1] G. H. Stokes, J. B. Evans, and S. M. Larson, “Near-earth asteroid search programs,” *Asteroids*, vol. 3, pp. 45–54, 2002.

[2] J. B. Evans, F. C. Shelly, and G. H. Stokes, “Detection and discovery of near-earth asteroids by the linear program,” *Lincoln Laboratory Journal*, vol. 14, no. 2, pp. 199–215, 2003.

[3] NASA. (2015) The NEAT survey. [Online]. Available: http://neat.jpl.nasa.gov/

[4] ——. (2014) The Catalina Sky Survey. [Online]. Available: http://www.lpl.arizona.edu/css/

[5] L. Denneau, R. Jedicke, T. Grav, M. Granvik, J. Kubica, A. Milani, P. Vereš, R. Wainscoat, D. Chang, F. Pierfederici *et al.*, “The pan-starrs moving object processing system,” *Pan*, vol. 125, no. 926, pp. 357–395, 2013.

[6] NASA. (2014) The NEOWISE survey. [Online]. Available: http://neo.jpl.nasa.gov/programs/neowise.html

[7] ——. (2015) A Near Earth Asteroid Census. [Online]. Available: http://tinyurl.com/bypjz4d

[8] M. Shao, B. Nemati, C. Zhai, S. G. Turyshev, J. Sandhu, G. Hallinan, and L. K. Harding, “Finding very small near-earth asteroids using synthetic tracking,” *The Astrophysical Journal*, vol. 782, no. 1, p. 1, 2014.

[9] T. Yanagisawa, A. Nakajima, K.-i. Kadota, H. Kurosaki, T. Nakamura, F. Yoshida, B. Dermawan, and Y. Sato, “Automatic detection algorithm for small moving objects,” *Publications of the Astronomical Society of Japan*, vol. 57, no. 2, pp. 399–408, 2005.

[10] P. S. Gural, J. A. Larsen, and A. E. Gleason, “Matched filter processing for asteroid detection,” *The Astronomical Journal*, vol. 130, no. 4, p. 1951, 2005.

[11] J. Kubica, J. Masiero, A. W. Moore, R. Jedicke, and A. Connolly, “Variable kd-tree algorithms for spatial pattern search and discovery,” *Robotics Institute*, p. 253, 2005.

[12] J. Kubica, A. Moore, A. Connolly, and R. Jedicke, “A multiple tree algorithm for the efficient association of asteroid observations,” in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. ACM, 2005, pp. 138–146.

[13] J. Kubica, L. Denneau, T. Grav, J. Heasley, R. Jedicke, J. Masiero, A. Milani, A. Moore, D. Tholen, and R. J. Wainscoat, “Efficient intra-and inter-night linking of asteroid detections using kd-trees,” *Icarus*, vol. 189, no. 1, pp. 151–168, 2007.

[14] P. Viola and W. M. Wells III, “Alignment by maximization of mutual information,” *International journal of computer vision*, vol. 24, no. 2, pp. 137–154, 1997.

[15] S. Larson, J. Brownlee, C. Hergenrother, and T. Spahr, “The catalina sky survey for neos,” in *Bulletin of the American Astronomical Society*, vol. 30, 1998, p. 1037.

[16] A. Drake, S. Djorgovski, A. Mahabal, E. Beshore, S. Larson, M. Graham, R. Williams, E. Christensen, M. Catelan, A. Boattini *et al.*, “First results from the catalina real-time transient survey,” *The Astrophysical Journal*, vol. 696, no. 1, p. 870, 2009.