ROAD: Rapid Optical Asteroid Detection

by

Julian Brown

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2016

Author ..... Signature redacted .....
Department of Electrical Engineering and Computer Science
May 20, 2016

Certified by ..... Signature redacted .....
Kerri L. Cahoy
Assistant Professor
Thesis Supervisor

Certified by ... Signature redacted .....
Benjamin F. Lane
Group Leader for Sensors and Imaging Systems
Thesis Supervisor

Accepted by .. Signature redacted .....
Christopher Terman
Chairman, Masters of Engineering Thesis Committee

ROAD: Rapid Optical Asteroid Detection

by

Julian Brown

Submitted to the Department of Electrical Engineering and Computer Science
on May 20, 2016, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

Here we present Tilt and the Three-Dimensional Discrete Radon Transform (3DRT), efficient image processing algorithms capable of identifying streaks at the detection limit in video of the night sky. Tilt and the 3DRT are asymptotically optimal algorithms for the blind search problem, which seeks to identify near-Earth asteroids of arbitrary position and velocity using ground-based optical systems. In the process of establishing the optimality of these algorithms, we formalize the blind search streak detection problem and survey several other state of the art algorithms that solve it: synthetic tracking, Fourier volume rendering, and the Approximate Discrete Radon Transform (ADRT). We also discuss the lessons learned from implementing a near-Earth asteroid detection system which demonstrated the 3DRT's capabilities by identifying five satellite streaks.

Thesis Supervisor: Kerri L. Cahoy
Title: Assistant Professor

Thesis Supervisor: Benjamin F. Lane
Title: Group Leader for Sensors and Imaging Systems

Acknowledgments

My deepest gratitude goes out to Benjamin Lane, Michael Sorensen, and Daniel Meiser, without whom this thesis would never have come to fruition. I'd also like to thank my thesis advisor, Kerri Cahoy for giving me the opportunity to work on this project. Finally, I'd like to thank my friends for their patience in waiting for this thesis to be release reviewed before probing me about my work.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the section, we present the motivations and challenges behind tracking near-Earth asteroids (NEAs). In Chapter 2, we summarize the advantages and disadvantages of the current state of the art in NEA detection. In Chapter 3, we present and analyze our new algorithms for NEA detection, Tilt and the 3DRT. In Chapter 4, we demonstrate the 3DRT in simulation and describe our implementation of a blind search asteroid detection system that runs in real time. Tilt and the 3DRT have the potential to significantly improve the performance of existing asteroid detection systems by increasing their sensitivity to NEAs. Our objective is that our algorithms will be an enabling technology for all kinds of asteroid related advancements, from space mining to impact avoidance.

1.1 Motivations

Improving our understanding of NEAs is important, not only for avoiding the potential destruction of an Earth impact, but also because most asteroids are as old as the solar system itself, so their composition can give us insight into the conditions present 4.6 billion years ago [11]. Observation can also provide a wealth of information about asteroid interaction, revealing important clues about the formation of Earth and other planets. NEAs are uniquely valuable, as they come close enough to Earth to allow for direct observation or even direct human interaction, as is proposed

in NASA's Asteroid Redirect Mission [13]. Additionally, it is estimated that 60 tons of space debris hits the Earth every day [4], so studying NEA composition can tell us about the cosmic dust contribution to the composition of Earth's surface.

Recently, asteroids have even become a potential avenue for economic growth. With decreasing availability of terrestrial resources, space mining is being looked to as a viable alternative. For example, Planetary Resources, a company founded in 2009, is actively working on the problem of NEA mining. Planetary Resources has set its sights on asteroids containing large quantities of water and rare metals like platinum [7]. A primary motivation for space mining is the prohibitively high cost of sending things to space, which may make it worthwhile to construct equipment for large-scale space missions directly using resources from asteroids, without ever sending materials back to Earth for processing. A simple example is radiation shielding for manned space missions. The shielding can be made of water extracted from asteroids, drastically reducing the launch weight for a long-term human vessel.

Asteroids also present an existential risk for humanity. Earth is frequently bombarded by interplanetary projectiles. The scarring we associate with the face of our moon is absent on the Earth due to weathering and erosion. The life-giving properties of our home world, its atmosphere and liquid water, have weathered away most of the remnants of past impacts. But a few, larger and more recent impact craters have survived well enough for us to observe, such as Arizona's 50,000 year old Meteor Crater depicted in Figure 1-1 or the Chicxulub crater associated with the Cretaceous-Paleogene mass extinction 66 million years ago [10]. These craters indicate the incredible destructive power of large asteroids. A large enough asteroid could easily wipe out all of humanity. But even smaller asteroids could pose a significant threat by causing damage to vital infrastructure or landing in a heavily populated area. Overall, it's estimated the average American is just as likely to die from an asteroid or comet impact as from a plane crash [3]. Asteroids present a very real existential threat and tracking them is the first step towards mitigating that threat.

Figure 1-1: Meteor Crater, an impact site located in northern Arizona. Source: `meteorcrater.com` - April 21, 2016

1.2 Modern Asteroid Detection

As depicted in Figure 1-3, the last 30 years have seen an exponential growth in NEA detections. The steep increase can be largely attributed to the Spaceguard goal, a congressional mandate that NASA find 90% of all NEAs larger than 1 km by 2008 and 90% of all NEAs larger than 140 m by 2020 [12]. Hundreds of millions of dollars have already been spent on systems capable of detecting NEAs. Even so, without upgrades to existing systems, we will likely fall short of the 2020 Spaceguard goal.

Modern asteroid detection is dominated by ground-based, optical systems. A notable exception is the Wide-field Infrared Survey Explorer (WISE) satellite, equipped with an infrared telescope [8]. When the satellite ran out of coolant in September 2010, the project was renamed to NEOWISE. NEOWISE has found a considerable number of NEAs, including many optically dark objects ground-based systems have trouble detecting due to the absorptive properties of Earth's atmosphere in the infrared. The transmission spectrum of Earth's atmosphere is shown in Figure 1-2. Despite NEOWISE's advantage, ground-based optical systems have found an order of magnitude more NEAs. Specifically, the Lincoln Near Earth Asteroid Research (LINEAR) program, the Catalina Sky Survey (CSS), and the Panoramic Survey Telescope and Rapid Response System (Pan-STARRS) are all ground-based opti-

cal systems and cumulatively account for the vast majority of all NEA detections to date. Their detection capabilites are shown in Figure 1-4.

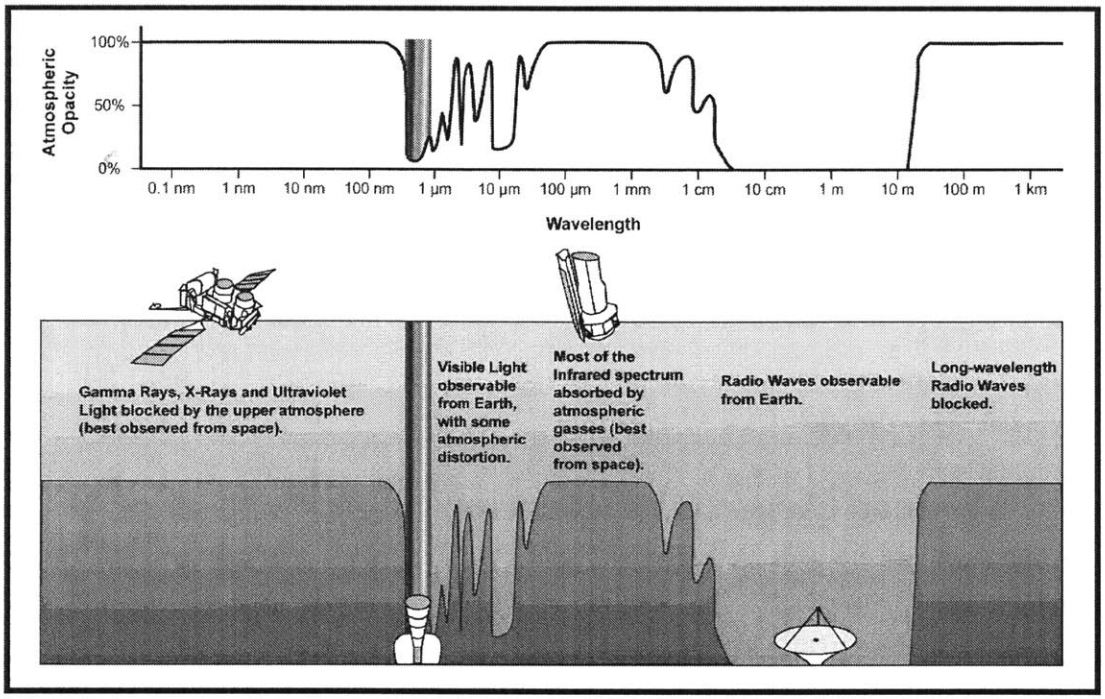


Figure 1-2: Transmission spectrum of Earth's astmosphere. Whereas much of the infrared spectrum is absorbed by the atmosphere, visible light is readily transmitted. Source: `http://gsp.humboldt.edu/olm_2015/Courses/GSP_216_Online/lesson2-1/atmosphere.html` - May 19, 2016

Ground-based optical systems have been largely successful due to the availability of environments devoid of stray light, and the affordability of full-sky coverage. But lots of coverage means lots of image data. The NEA detection algorithms currently in use are severely limited due to computational constraints in processing such a large amount of image data [11]. As a result, dim or fast-moving NEAs have been passing undetected.

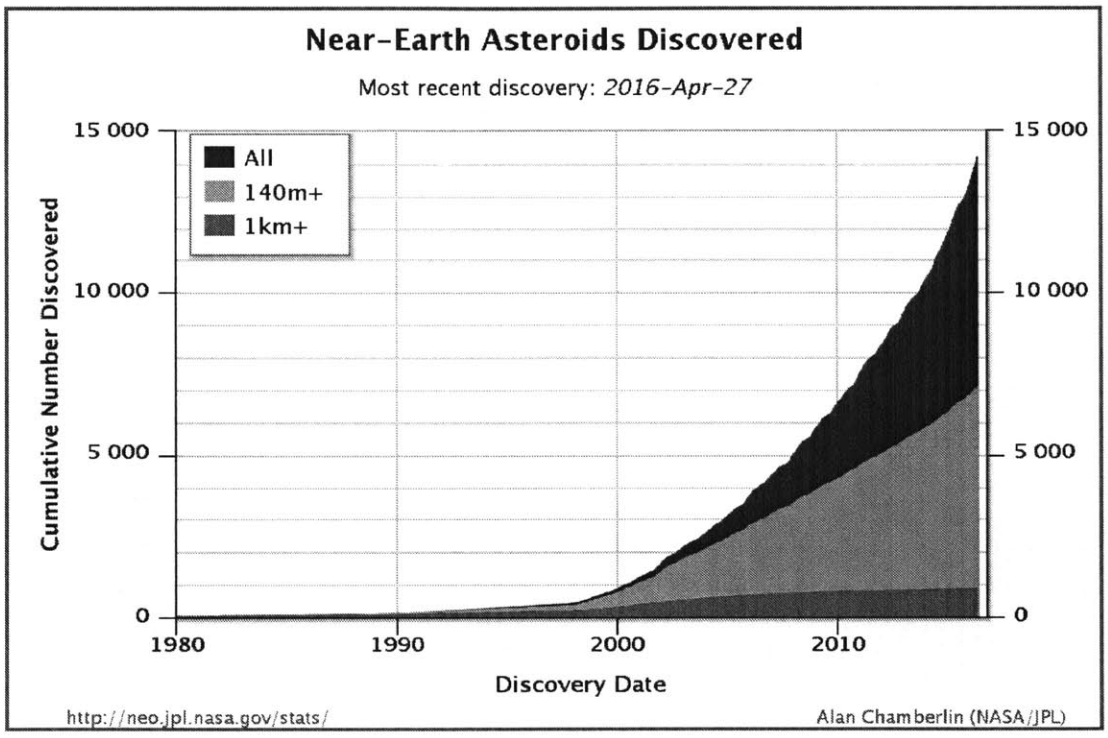


Figure 1-3: Near-Earth Asteroids discovered to date. Over 90% of NEAs larger than 1 km have already been discovered. Smaller NEAs are being discovered at an increasing rate. Source: `neo.jpl.nasa.gov/stats/` - April 30, 2016

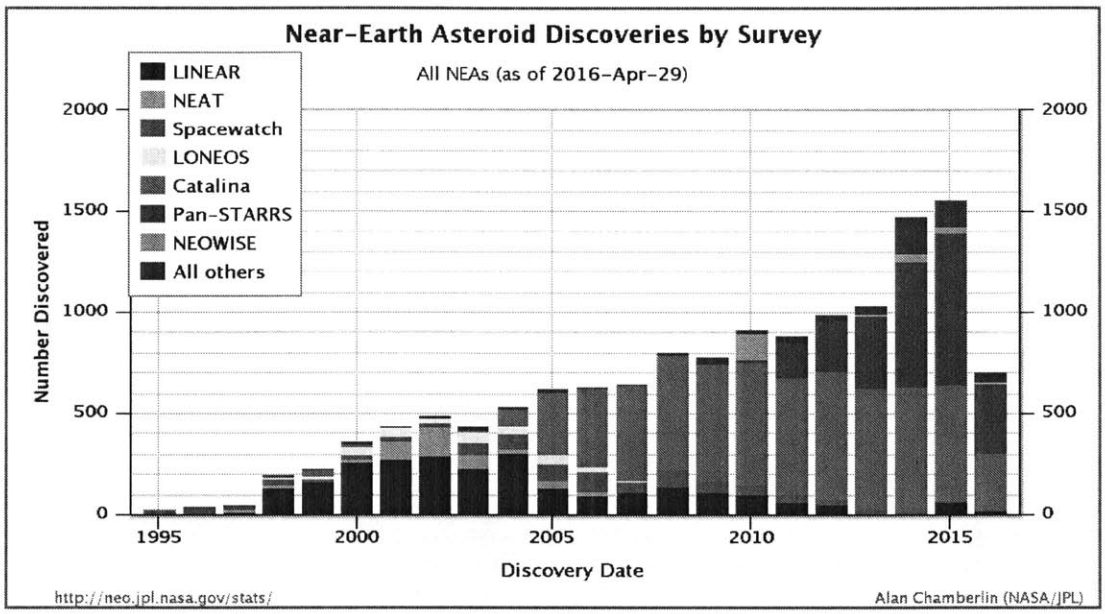


Figure 1-4: The most successful NEA detection programs to date are the Catalina Sky Survey (CSS) and the Panoramic Survey Telescope and Rapid Response System (Pan-STARRS). Source: `neo.jpl.nasa.gov/stats/` - April 30, 2016

1.3 Problem Statement

Optical asteroid detection is the problem of identifying asteroids in video of the night sky. The problem is composed of three subproblems, which can be summarized as follows:

1. Correct for camera motion relative to the stars. Camera stabilization prevents stars from streaking and being misidentified as asteroids. This includes correcting for sidereal motion, the motion of the Earth against the stars.

2. Remove stationary bright objects, including stars and galaxies. When searching for asteroids in images of the night sky, stars are the primary source of error. A sufficiently bright star can cause multiple false detections.

3. Identify streaks running through the image. Streaks are potential asteroids crossing the starfield. Figure 1-5 shows three different kinds of streaks, which should all be identified as asteroid candidates.

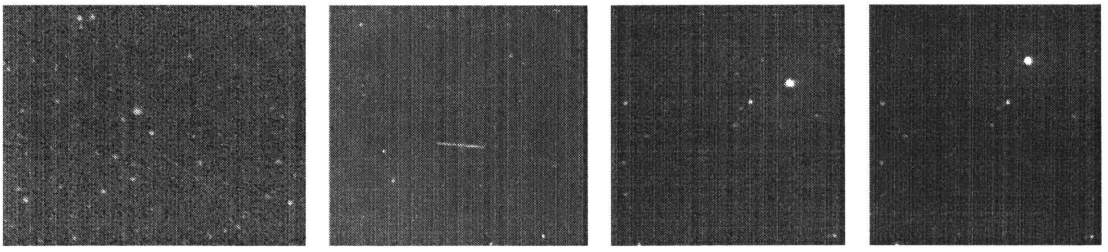


Figure 1-5: Three different kinds of streaks. In the first image, the asteroid enters the field-of-view in the lower right and exits in the upper left, creating a streak fully contained within a single frame. In the second image, the asteroid smears across this frame, but may be visible across multiple frames. In the third and fourth images, which were taken two minutes apart, the brightest star is actually an asteroid moving slowly enough to appear frozen in any individual frame, but can be seen to move over successive frames. These images were taken during testing of our blind search asteroid detection system's camera and telescope.

Streak identification is the primary focus of this thesis, as it poses the greatest computational challenge. This may seem counter-intuitive, as the human eye is extremely adept at picking out lines and moving objects, so the task appears trivial at

first. However, automating the task requires breaking the process up into manageable steps and humans would be hard-pressed to describe exactly how their visual processes operate. To make matters worse, even the human eye fails to identify streaks close to the detection limit, as shown in Figure 1-6.

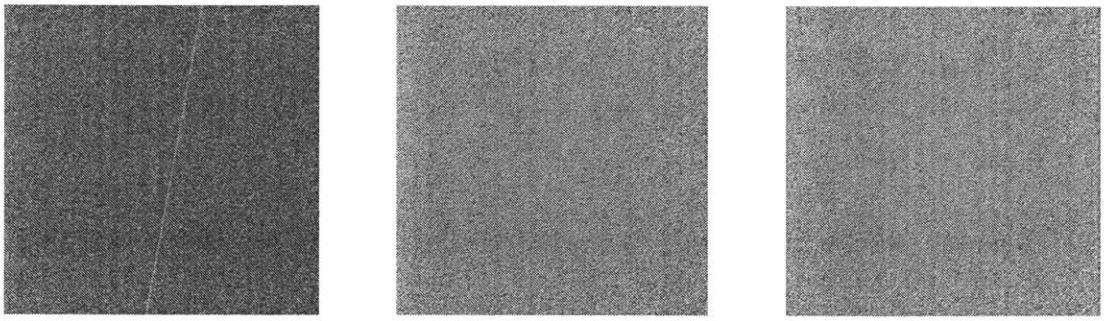


Figure 1-6: Three simulated images containing the same Gaussian noise and the same streak running from top to bottom. The signal strength of a streak is defined as how many standard deviations its integral is above the mean integral for lines that contain only noise. The first image's streak is an 80 sigma signal, the second 20 sigma, and the third 5 sigma, barely above the noise floor and undetectable to the human eye.

Chapter 2

State of the Art

In this chapter, we outline the current state of the art in asteroid detection. We begin by formalizing the problem of blind search streak detection. We then present three different streak detection techniques: synthetic tracking, fourier volume rendering, and the approximate discrete radon transform (ADRT). We also discuss the pros and cons of each algorithm in terms of runtime and noise properties.

2.1 Blind Search Streak Detection

The problem of blind search streak detection may be formalized as follows. We are given an $N \times N \times N$ cube of pixel values known as an image stack, wherein the first two dimensions are the image dimensions and the third dimension is time. A cube is representative for the blind search streak detection problem, as image stack dimensions are typically within a small multiplicative factor of each other, so zero-padding to a cube does not affect the asymptotic runtime or noise properties of any of the algorithms analyzed in this paper. The cube contains per-pixel Gaussian noise of standard deviation σ and a streak approximately one pixel wide, which enters on one face of the cube and exits on another. This model represents a detector system where all patterns and noise regularities have already been removed, so all that remains is detector and sky background noise. Our goal is to identify the line along which the streak lies.

2.2 Synthetic Tracking [11]

Synthetic tracking is the algorithmically naive approach to the blind search streak detection problem. It computes every possible line integral through the image stack directly, by summing together the pixels values making up each line. It then checks whether the line integral is too bright to consist only of noise and must therefore contain the streak. There are N^2 pixels the streak may enter or exit on a given face of the cube and $\binom{6}{2} = 15$ combinations of faces, which gives a total of $15N^4$ line integrals for synthetic tracking to compute. Every line through the cube consists of at most N pixels, so directly computing each line integral requires $O(N)$ operations. Synthetic tracking therefore requires $O(N^5)$ operations to solve the problem of blind search streak detection. The trade-off for this relatively large runtime is that synthetic tracking has the best possible noise properties. The line integral running along the streak contains the streak's N pixels and no others. The noise in the line integral is therefore $O(\sigma\sqrt{N})$, the lowest value possible.

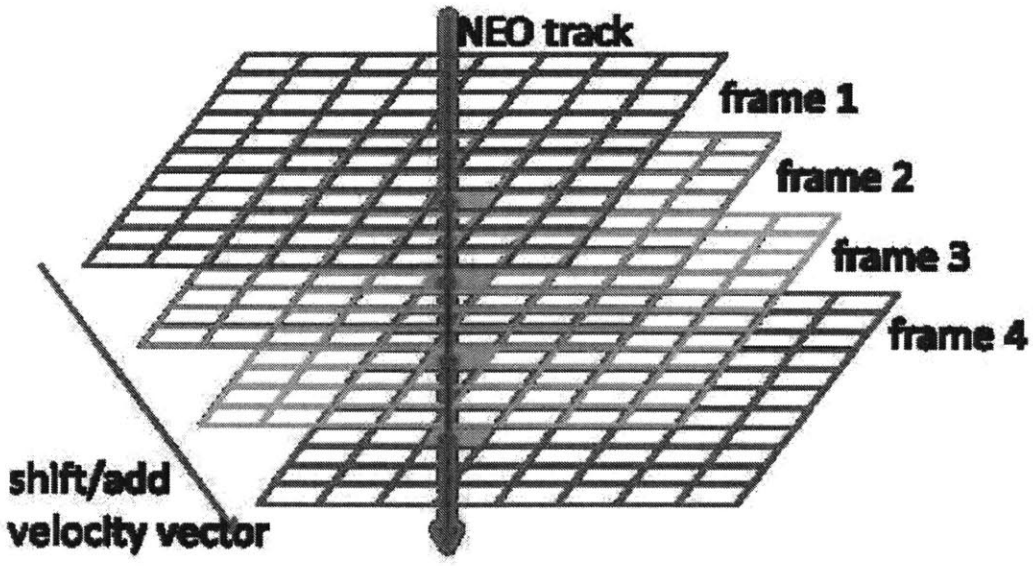


Figure 2-1: Synthetic tracking aligns near-Earth Object (NEO) streaks by shifting consecutive image frames. Image taken from Zhai's "Detection of a Faint Fast-moving Near-Earth Asteroid Using Synthetic Tracking Technique" [14].

2.3 Fourier Volume Rendering [6]

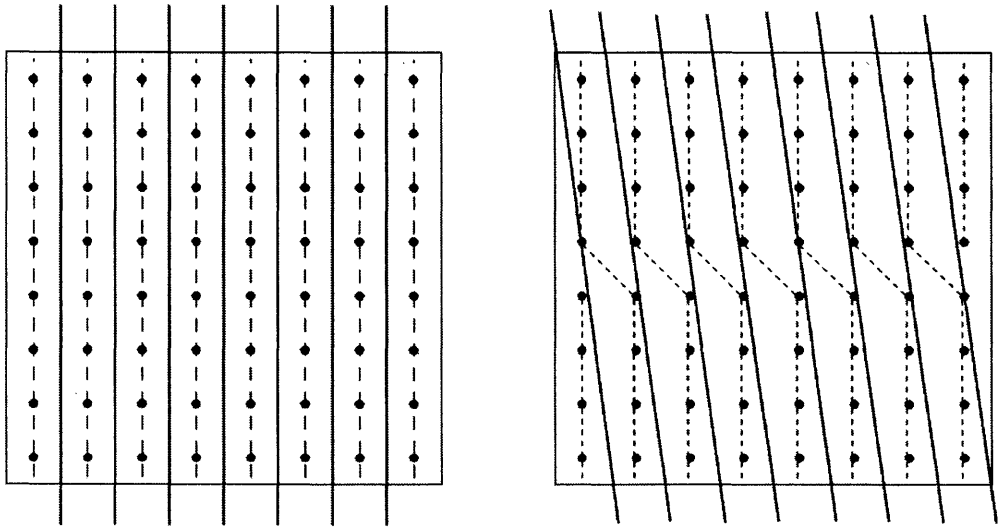
Fourier Volume Rendering is a technique that operates on a frequency-domain representation of the data set, allowing us to use a fast Fourier transform to compute line integrals in log time. The technique relies on The Fourier projection-slice theorem, which states that "the inverse transform of a slice extracted from the frequency domain representation of a volume yields a projection of the volume in a direction perpendicular to the slice." In other words, Fourier Volume Rendering first transforms the image stack to the frequency-domain, then extracts a 2D slice of the cube. Running the inverse Fourier transform on this slice gives us all of the line integrals perpendicular to the slice. Converting the image stack to the frequency domain is an $O(N^3 \log N)$ time operation that need only be done once, while each of the slices takes $O(N^2 \log N)$ time to compute $O(N^2)$ line integrals. In order to compute all $O(N^4)$ line integrals, we need to use $O(N^2)$ slices, which gives a total runtime of $O(N^4 \log N)$. This is a significant asymptotic speedup over synthetic tracking's $O(N^5)$ runtime.

But Fourier Volume Rendering is not a perfect solution. Operating in the frequency-domain introduces a large constant to the runtime by way of additional operations required for the fast Fourier transform and the need to use complex numbers instead of integers. Additionally, while the noise properties should be similar to those of synthetic tracking, Fourier Volume Rendering does not produce exact line integrals when the extracted slice must be interpolated, which occurs for the majority of slices. The noise introduced by slice interpolation may make Fourier Volume Rendering unfit to solve the problem of blind search streak detection.

2.4 Approximate Discrete Radon Transform [2]

The approximate discrete radon transform (ADRT) is able to compute every line integral through an $N \times N$ image in $O(N^2 \log N)$ operations. The key insight at the core of the ADRT is that adjacent line integrals share subsegments. This concept is illustrated in Figure 2-2. Whereas synthetic tracking recomputes shared subsegments

for every line integral, the ADRT computes shared subsegments only once, as seen in Figure 2-3. The computational savings are enough that the ADRT computes each line integral in $O(\log N)$ time, which is the same complexity as Fourier Volume Rendering. However, since the ADRT operates on two-dimensional data sets, it cannot be used directly on the image stack. A simple fix is to sum the images along the time dimension, returning one image equivalent to a single exposure spanning the entire duration of the image stack. But this adds a significant amount of noise, as the ADRT is now summing over N^2 pixels worth of noise instead of only the N original pixels containing the streak. The noise in the line integral is therefore $\sigma\sqrt{N^2}$ or σN, which may be a worthwhile tradeoff for the $O(N^2 \log N)$ runtime of the ADRT.



(a) Angle θ_1. (b) Angle θ_2.

Figure 2-2: Adjacent line integrals have common subsegments. The ADRT takes advantage of this overlap to compute line integrals in log time. Image taken from Brady's "A Fast Discrete Approximation Algorithm for the Radon Transform" [2].

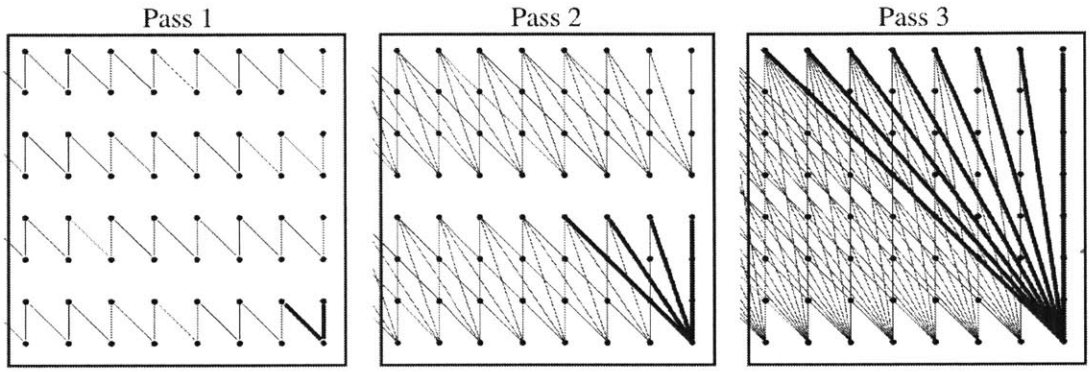


Figure 2-3: The ADRT creates N^2 new line segments in each pass by combining two pixels for the first pass and two line segments from the previous pass for every pass other than the first. Since the line segments double in length with each pass, only $\log N$ passes are needed to complete the ADRT. With $\log N$ passes and N^2 additions in each pass, the ADRT runs in $O(N^2 \log N)$ time. Image taken from Brady's "A Fast Discrete Approximation Algorithm for the Radon Transform" [2].

Chapter 3

A New Approach

The current state of the art algorithms for blind search streak detection all have undesirable aspects that make them computationally untenable: synthetic tracking is asymptotically slow, Fourier volume rendering has a large constant factor in its runtime, and the ADRT is noisy. In this chapter, we outline two algorithms, 3DRT and Tilt, which are asymptotically fast, have minimal constant factors in their runtimes, and have optimal noise properties. The new algorithms work by generalizing the ADRT to three dimensions in two different ways.

3.1 Tilt

Tilt begins by selecting one row from each of two faces of the image stack. Tilt then computes the ADRT of the discrete slice that lies between the selected rows, zero-padding the rectangular slice if the faces are not opposite. This concept is illustrated in Figure 3-1. Tilt repeats this procedure for all N^2 pairs of rows, tilting the slice through the cube. Tilt does this for all 15 pairs of faces, resulting in a total of $15N^2$ ADRTs, each of which computes N^2 line integrals in $O(N^2 \log N)$ time. Therefore, Tilt computes all $15N^4$ line integrals in $O(N^4 \log N)$ time. Each line integral has the same value as if it had been computed by synthetic tracking, and therefore has the same optimal noise value of $O(\sigma\sqrt{N})$.

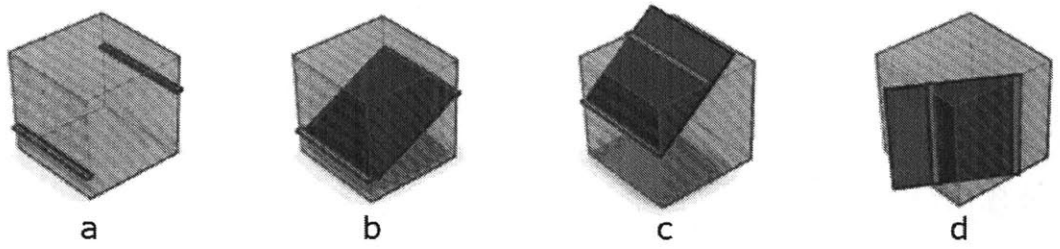


Figure 3-1: In a, Tilt selects two parallel rows depicted in orange. In b, Tilt runs the ADRT on the plane, depicted in red, uniquely defined by the rows from a. In c, the rows occupy adjacent faces of the cube, so the plane juts out of the image stack and must be zero-padded. In d, the rows must be vertical in order to be parallel for the given faces.

3.2 3DRT

The 3D ADRT, or 3DRT, is similar to Tilt, in that it computes the same $15N^2$ ADRTs. The difference being that the 3DRT takes advantage of the overlap between adjacent ADRTs. In contrast to Tilt, the 3DRT selects a row from only one of the faces and computes in parallel the N ADRTs corresponding to the N rows of the other face as illustrated in Figure 3-2. This operation is repeated for each of the N rows of the first face, thereby calculating all N^2 ADRTs for that pair of faces and $15N^2$ ADRTs over all pairs of faces. The noise properties and asymptotic runtime of the 3DRT are the same as Tilt's, but the 3DRT has a small constant performance boost at the cost of requiring N^3 memory to store N ADRTs instead of Tilt's N^2 memory to store a single ADRT at a time. While the 3DRT performs fewer operations than Tilt, each operation takes longer due to poor locality, making Tilt the better algorithm in practice.

It's worth mentioning that the technique used to create the 3DRT can be extended to create a 4DRT, which calculates all N^2 ADRTs for a given pair of faces at once. The benefit is that the 4DRT has an even smaller constant runtime factor than the 3DRT. However, storing N^4 values in fast-access memory is not viable in practice. For $N = 1000$, Tilt requires Megabytes of active memory, the 3DRT requires Gigabytes, and the 4DRT requires Terabytes.

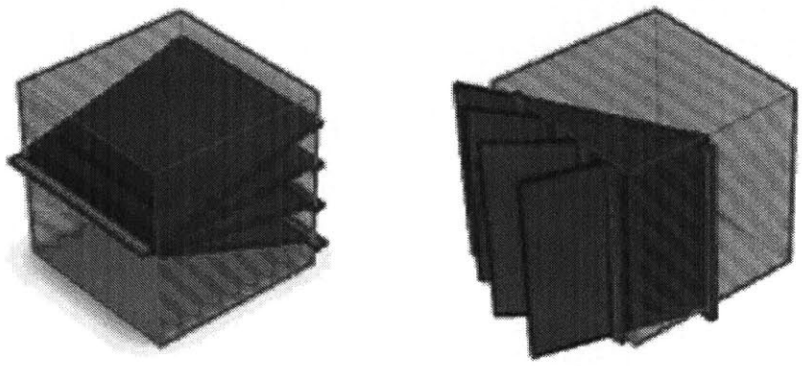


Figure 3-2: Instead of selecting one row for each face, the 3DRT selects one row for the first face and simultaneously computes the N ADRTs from that row to every row of the second face.

Table 3.1: Algorithm Complexities and Noise Properties

Algorithm	Time	Space	Noise
Synthetic Tracking	$O(N^5)$	$O(1)$	$O(\sigma\sqrt{N})$
Fourier Volume Rendering	$O(N^4 \log N)$	$O(N^2)$	$O(\sigma\sqrt{N})$
ADRT	$O(N^2 \log N)$	$O(N^2)$	$O(\sigma N)$
Tilt	$O(N^4 \log N)$	$O(N^2)$	$O(\sigma\sqrt{N})$
3DRT	$O(N^4 \log N)$	$O(N^3)$	$O(\sigma\sqrt{N})$

Chapter 4

Results

In this chapter, we discuss testing of the 3DRT. The 3DRT was implemented in Python for small-scale simulation testing and CUDA for real-world testing. We encountered many issues when working with the real-world data and developed several engineering solutions: camera stabilization was done with star tracking, asteroids were substituted with satellites, and bright streaks were resolved by subsampling. In the end, the 3DRT was able to identify 5 geo-synchronous satellites moving through an image stack.

4.1 Simulation

Prior to real-world testing, we wanted to demonstrate the 3DRT in simulation. We started by generating an empty $16 \times 16 \times 16$ image stack. We then inserted a randomly generated streak into the image stack. Streaks consisted of $10,000$ counts, distributed evenly through the time axis of the image stack and with 0.2 pixel standard deviation Gaussian noise in the x and y axes. Then, the 3DRT was used to infer the streak's path through the image stack. The 3DRT needed to directly detect $2,500$ counts and the inferred path needed to overlap at least $5,000$ counts in order for the test to be considered a success. A representative run of the simulation is depicted in Figure 4-1. The $2,500$ count minimum for direct detection is motivated by the worst case scenario of a vertical streak positioned precisely at the intersection of four pixels.

Such a streak would be detected as four $2,500$ count line integrals. The simulation was run $10,000$ times without any failures.

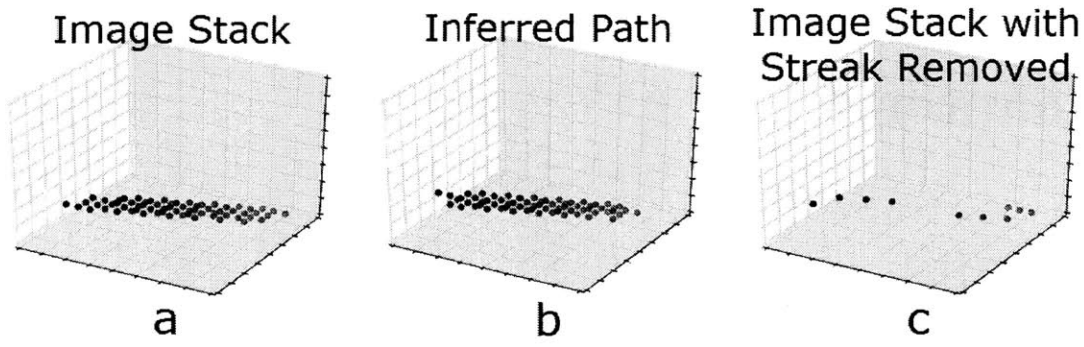


Figure 4-1: The 3DRT identifies a streak's path through the image stack. In a, a randomly generated streak runs through the image stack. In b, the streak's path is shown as inferred by the 3DRT. In c, the image stack with the pixels that lie along the inferred path have been removed, demonstrating the accuracy of the 3DRT's inference.

4.2 Testing

Once we verified the 3DRT in simulation, it was time to implement a full-scale system and work with real-world data. The 3DRT was reimplemented in CUDA and a telescope rig was set up with a PCO sCMOS camera and a Canon EF 600mm f/4 telephoto lens. The rig was quite powerful, able to detect 12th magnitude stars in a single frame. But this came at the cost of a relatively small field-of-view (FOV): each frame was only 1.6 degrees along the horizontal. The small FOV and the rig's lack of mechanical stabilization meant the sidereal motion of the Earth was a serious issue. Over the duration of an image stack, the camera would reorient to look at a different starfield. To remedy this, we implemented a lost-in-space star tracking algorithm using $300,000$ stars from the Tycho-2 Catalog. However, due to the relatively large number of stars required for small FOV star tracking, the Pyramid algorithm [9] would have spent hours identifying a single frame and algorithms that use larger patterns of stars, like astrometry.net [5], would require unwieldy catalogs. By modifying the Pyramid algorithm's cross-referencing method to use hash tables, we managed to get

the star tracker working in real time. We used the attitude estimates for each frame to digitally stabilize the image stack, correcting for sidereal motion as well as any shaking of the rig.

The rig's immotility and small FOV, combined with the infrequency of NEAs meant asteroids were an impractical streak source for testing. Fortunately, geosynchronous (GEO) and low-Earth-orbiting (LEO) satellites are a perfect substitute for asteroids: they're relatively common and typically dim. Pointing the rig towards the equatorial plane resulted in a suitable view of geostationary orbit. The apparatus worked a little too well, capturing multiple bright GEO satellites, which have visible streaks seen in Figure 4-2. The The 3DRT was overloaded by the detection of thousands of line integrals that only partially intersected these bright streaks.

The 3DRT is too sensitive: extremely bright objects need to be filtered out in a preprocessing step. Luckily, this can be easily accomplished by down-sampling the image stack by a factor of two in each dimension, running the 3DRT on the smaller image stack, and removing detected streaks from the larger image stack as in Figure 4-1. This down-sampling trick is the application of a more general technique known as Pyramid image processing [1]. With this change in place, the 3DRT successfully extracted all five GEO satellite streaks.

4.3 Contributions

The primary contribution of this thesis is the pair of algorithms it presents, Tilt and the 3DRT. These algorithms constitute the best solution to the blind search streak detection problem so far, outperforming all other algorithms currently in use. This thesis also constitutes a survey of the current state of the art in NEA detection and formalizes the problem of blind search streak detection, simplifying comparison of the competing algorithms. Finally, this thesis contributes lessons learned from engineering a working implementation of blind search streak detection using the 3DRT.

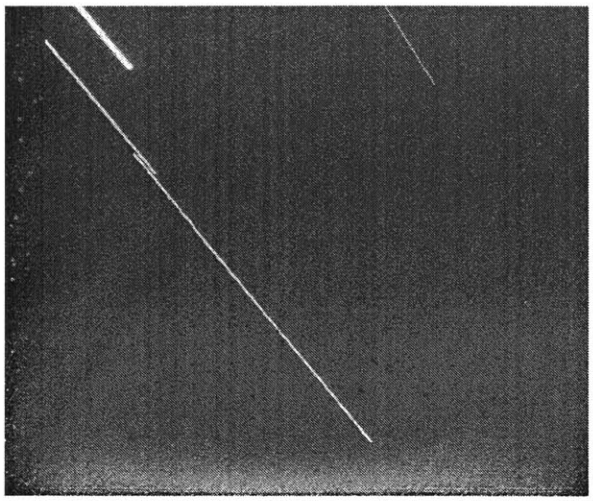


Figure 4-2: Five GEO satellite streaks are visible in this flattened image stack.

4.4 Future Work

Due to time constraints, we were only able to conduct proof-of-concept testing for the 3DRT. In the future, we will conduct experiments with both Tilt and the 3DRT. Future testing will also be done on a longer time scale to allow for use of NEAs instead of satellites. Additionally, we plan on more fully characterizing the practical performance of Tilt and the 3DRT relative to the state of the art algorithms by comparing runtimes and sensitivity on representative data sets.

Bibliography

[1] Edward H Adelson, Charles H Anderson, James R Bergen, Peter J Burt, and Joan M Ogden. Pyramid methods in image processing. *RCA engineer*, 29(6):33–41, 1984.

[2] Martin L Brady. A fast discrete approximation algorithm for the radon transform. *SIAM Journal on Computing*, 27(1):107–119, 1998.

[3] Clark R Chapman, David Morrison, et al. Impacts on the earth by asteroids and comets: assessing the hazard. *Nature*, 367(6458):33–40, 1994.

[4] Chester S. Gardner, Alan Z. Liu, D. R. Marsh, Wuhu Feng, and J. M. C. Plane. Inferring the global cosmic dust influx to the earth's atmosphere from lidar observations of the vertical flux of mesospheric na. *Journal of Geophysical Research: Space Physics*, 119(9):7870–7879, 2014. 2014JA020383.

[5] Dustin Lang, David W Hogg, Keir Mierle, Michael Blanton, and Sam Roweis. Astrometry. net: Blind astrometric calibration of arbitrary astronomical images. *The astronomical journal*, 139(5):1782, 2010.

[6] Marc Levoy. *Volume rendering using the Fourier projection-slice theorem.* Computer Systems Laboratory, Stanford University, 1992.

[7] Chris Lewicki, Peter Diamandis, Eric Anderson, Chris Voorhees, and Frank Mycroft. Planetary resourcesâĂŤthe asteroid mining company. *New Space*, 1(2):105–108, 2013.

[8] Amy Mainzer, T Grav, J Bauer, J Masiero, RS McMillan, RM Cutri, R Walker, E Wright, P Eisenhardt, DJ Tholen, et al. Neowise observations of near-earth objects: Preliminary results. *The Astrophysical Journal*, 743(2):156, 2011.

[9] Daniele Mortari, Malak A Samaan, Christian Bruccoleri, and John L Junkins. The pyramid star identification technique. *Navigation*, 51(3):171–183, 2004.

[10] Peter Schulte, Laia Alegret, Ignacio Arenillas, José A Arz, Penny J Barton, Paul R Bown, Timothy J Bralower, Gail L Christeson, Philippe Claeys, Charles S Cockell, et al. The chicxulub asteroid impact and mass extinction at the cretaceous-paleogene boundary. *Science*, 327(5970):1214–1218, 2010.

[11] Michael Shao, Bijan Nemati, Chengxing Zhai, Slava G Turyshev, Jagmit Sandhu, Gregg Hallinan, and Leon K Harding. Finding very small near-earth asteroids using synthetic tracking. *The Astrophysical Journal*, 782(1):1, 2014.

[12] GH Stokes, DK Yeomans, WF Bottke, SR Chesley, JB Evans, RE Gold, AW Harris, D Jewitt, TS Kelso, RS McMillan, et al. Study to determine the feasibility of extending the search for near-earth objects to smaller limiting diameters. *Report of the Near-Earth Object Science Definition Team*, page 21, 2003.

[13] Nathan Strange, Damon Landau, G Lantoine, T Lam, M McGuire, L Burke, M Martini, and J Dankanich. Overview of mission design for nasa asteroid redirect robotic mission concept. In *33rd International Electric Propulsion Conference, The George Washington University, Washington, DC*, 2013.

[14] Chengxing Zhai, Michael Shao, Bijan Nemati, Thomas Werne, Hanying Zhou, Slava G Turyshev, Jagmit Sandhu, Gregg Hallinan, and Leon K Harding. Detection of a faint fast-moving near-earth asteroid using the synthetic tracking technique. *The Astrophysical Journal*, 792(1):60, 2014.