MITIGATING THE EFFECTS OF BOOM OCCLUSION ON AUTOMATED AERIAL REFUELING THROUGH SHADOW VOLUMES

THESIS

Zachary C. Paulson, 2nd LT, USAF

AFIT-ENG-MS-18-M-051

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENG-MS-18-M-051

MITIGATING THE EFFECTS OF BOOM OCCLUSION ON AUTOMATED AERIAL REFUELING THROUGH SHADOW VOLUMES

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Computer Science

Zachary C. Paulson, B.S.C.S

2nd LT, USAF

March 2018

AFIT-ENG-MS-18-M-051

MITIGATING THE EFFECTS OF BOOM OCCLUSION ON AUTOMATED AERIAL REFUELING THROUGH SHADOW VOLUMES

THESIS

Zachary C. Paulson, B.S.C.S
2nd LT, USAF

Committee Membership:

Dr. Scott Nykl
Chair

Dr. Douglas Hodson
Member

Dr. Robert Leishman
Member

Abstract

In flight refueling of Unmanned Aerial Vehicles (UAVs) is critical to the United State's Air Force (USAF). However, the large communication latency between a ground-based operator and his/her remote UAV makes docking with a refueling tanker unsafe. This latency may be mitigated by leveraging a tanker-centric stereo vision system. The vision system observes and computes an approaching receiver's relative position and orientation offering a low-latency, high frequency docking solution. Unfortunately, the boom – an articulated refueling arm responsible for physically pumping fuel into the receiver – occludes large portions of the receiver especially as the receiver approaches and docks with the tanker. The vision system must be able to compensate for the boom's occlusion of the receiver aircraft. We present a novel algorithm for mitigating the negative effects of boom occlusion in stereo-based aerial environments. Our algorithm dynamically compensates for occluded receiver geometry by transforming the occluded areas into shadow volumes. These shadow volumes are then used to cull hidden geometry that is traditionally consumed, in error, by the vision processing and point registration pipeline. Our algorithm improves computer-vision pose estimates by an average of 74% over a naïve approach without shadow volume culling.

Acknowledgements

Special thanks to our sponsor, AFRL/RQ for their support in this research endeavor.

I would like to thank my family, friends, and mentors for supporting me throughout my military career.

Zachary C. Paulson

Table of Contents

Page

List of Figures

Figure Page

List of Tables

MITIGATING THE EFFECTS OF BOOM OCCLUSION ON AUTOMATED AERIAL REFUELING THROUGH SHADOW VOLUMES

I. Introduction

1.1 Overview

The United States Air Force must be able to secure and maintain global reach, superiority, and power to achieve its Core Missions. The Air Force's dependence on Unmanned Aerial Vehicles (UAVs) to achieve global reach has rapidly increased and UAVs have become a critical system for conducting reconnaissance. In the past, the Air Force was able to maintain global reach through aerial refueling, allowing our fleet of aircraft to maintain a global presence. However, there is currently no capability for UAVs to be refueled mid-flight severely degrading the Air Forces ability to achieve these core missions. The latency between an UAV operator and the UAV is too significant to allow for such an operation. To overcome this limitation the United States Air Force has been working on the development of a capability called automate aerial refueling (AAR) that will guide the UAV into the refueling threshold through relative positioning. Research has been conducted to show that the use of Global Positions Systems (GPS) can successfully perform relative positioning between aircraft, however, in a combat environment GPS is not a reliable resource as it can be denied by opposing forces or lost. Our solution proposes an automated system using a stereo computer vision pipeline to calculate the required 6 degrees of freedom (6DoF) measurements for the receiver aircraft relative to the tanker. Testing a proposed system often requires multiple flight tests which can only be performed after meeting strin-

gent safety requirements. To overcome such obstacles our research employes a 3D Virtual World (3DVW) simulation. This simulation models a hypothetical tanker-mounted stereo vision system able to observe an approaching receiver. The tanker and receiver aircraft are represented via geometrically realistic aircraft models and follow realistic flight paths representative of common refueling approaches. Geometrically realistic aircraft models are of an adequate fidelity to be representative of real world experiments. Previous research has shown that stereo computer vision can produce centimeter level accuracy. While highly useful, these simulations did not include a model of the tanker's refueling boom. The tanker's refueling boom obstructs the system's view of the receiver presenting problems in the stereo computer vision pipeline.

1.2 Problem Statement

Mitigate the effects of boom occlusion on the stereo computer vision system to produce realistic and effective automated aerial refueling capability in GPS denied environments with minimal aircraft modification.

1.3 Research Goals and Hypothesis

- Establish baseline of stereo computer vision relative 6DoF estimations with no boom mitigation techniques
- Improve stereo computer vision relative 6DoF estimation accuracy through the use of a dynamically created reference model
- Produce simulation results to compare with flight test data superimposed with a model of the tanker's boom

1.4 Approach

Experiments to quantify reference model performance are executed in the 3DVW. A tanker-centric stereo vision system captures synthetic imagery of an approaching receiver. The receiver follows a flight path within the operational limits of a C12. This flight path provides the truth data to compare against the 6DoF estimation produced by the computer vision pipeline. Three sets of deterministic experiments are executed to provide comparisons between no boom mitigation techniques, a more accurate static reference model consisting of just the wings and nose, and a dynamic reference model generated through the use of shadow volumes.

1.5 Assumptions/Limitations

The 3DVW presented in this research has been compared with measured data validating that the model is an accurate reflection of the real world [4]. This research does account for the partial occlusion of the receiver resulting from the tanker's refueling boom. This research assumes a completely static boom with knowledge of its location and orientation. The propellers of the C-12 receiver are not modeled for simplification. The flight path used in all of the experiments was generated using BlueMax flight software however, the virtual world does not use a flight dynamics engine to simulate turbulence or other aerodynamic effects on either the receiver or the tanker and it's respective refueling boom. Virtual sensors in the 3DVW follow the pinhole camera model such that lens distortion was not present.

1.6 Research Contributions

- 3DVW for simulating AAR computer vision applications
- The use of collision detection to eliminate boom generated points in the sensed

point cloud

- 74% average improvement on 6DoF estimation versus no mitigation techniques

1.7 Thesis Overview

Chapter 2 provides fundamental concepts necessary to understand the AAR problem. It defines terminology associated with aerial refueling and discusses fundamental concepts for computer vision and shadow volumes. Chapter 2 also provides a review of previous work conducted on the AAR problem. Chapter 3 provides the methodology for this research including the computer vision pipeline, ray-plan collision detection, face adjacency information, silhouette detection, shadow volume generation and point inclusion. Chapter 4 discusses the results from the experiments comparing the various mitigation techniques. In chapter 5 we conclude by giving an overview of the current state of AAR while also recommending steps to further improve the results of this research.

II. Background

2.1 Automated Aerial Refueling

In flight refueling of aircraft is critical to the United State's Air Force (USAF). It allows the USAF to maintain global reach and air superiority. The Air Forces dependence on Unmanned Aerial Vehicles (UAVs) to achieve missions has rapidly increased. UAVs are primarily used for surveillance and reconnaissance but their role is expanding into combat areas. Thus, it is critical that we develop the capability to refuel UAVs mid-flight. However, the large communication latency between a ground-based operator and his/her remote UAV makes docking with a refueling tanker unsafe. This multi-second latency even prevents stateside UAV operators from controlling take off and landing of remotely located UAVs [5]. This latency may be mitigated by leveraging a tanker-centric stereo vision system. The vision system observes and computes an approaching receiver's relative position and orientation offering a low-latency, high frequency docking solution. This vision system provides the pose estimation as six degrees of freedom with three components for position and three for orientation. An AAR solution could eventually eliminate the need for boom operators and would push the USAF towards completely autonomous refueling tankers.

Automated Aerial Refueling while not a current capability is being planned for in the future. Thus, an approved flight path has already been defined [6]. The NATO standard states that a receiver aircraft's flight path will vary based on aircraft type and environmental conditions. However, the flight path states that the receiver should follow an approach that limits occlusion of the receiver aircraft [6]. This guideline maximizes the capability of a computer vision solution to estimate position and orientation. Unfortunately, the boom – an articulated refueling arm responsible for physically pumping fuel into the receiver – occludes large portions of the receiver

especially as the receiver approaches and docks with the tanker. An example of the boom refueling method can be seen in Figure 1. This occlusion can not be overcome by modifying the flight path or the NATO standard. Thus, the vision system must be able to compensate for the boom's occlusion of the receiver aircraft.

2.2 Computer Vision

Computer vision is the process of acquiring images of the external world, processing them and then analyzing them to produce data or information [7]. Humans can quickly analyze imagery to make inferences and comparisons. While research in artificial intelligence is making strides towards creating systems with human-like perception the capability is not yet achieved [8]. Thus, computer systems rely on recognition through corners, or colors, or other specific features. Humans also have the capability to perceive depth, while computer vision systems with a singular camera can calculate depth in certain circumstances [9, 10] it is much less straight forward than using a stereo vision system. A stereo vision system is able to estimate depths of object with much greater precision.

Stereo vision systems are designed to mimic human depth perception. These systems utilize epipolar geometry to compute the depth of objects contained within each camera's viewing frustum [11]. A stereo vision system with its corresponding epipolar geometry components can be seen in Figure 2. The optical centers of both cameras are represented by O_1 and O_2. They are separated by a baseline that must be measured. The epipoles, e_1 and e_2, of the left and right cameras respectively. An epipole is defined as the intersection point between the baseline and the image plane of the camera. The epipole is the optical center of the other camera. The object of concern is represented by q and the projections of that object onto each image plane is represented by p_1 and p_2. The epipolar lines, l_1 and l_2, is the line that represents the

Figure 1. Boom Refueling Method [1]

intersection of the epipolar plane with each image plane respectively. The essential matrix contains the epipolar information necessary to relate corresponding image points. The epipolar constraint is the idea that q generates p_1 for the left camera and the corresponding point p_2 in the right camera must lie somewhere on the epipolar line. This greatly reduces the search for corresponding image points. Epipolar geometry thus allows for the 3D coordinates of q to be calculated with respect to optical center of the predetermined primary camera.

2.3 Pinhole Camera Model

The pinhole camera model is the ideal model of a camera. In the pinhole model the camera's aperture is a singular point; thus all rays pass through the same center point of the camera. Our research assumes a pinhole camera model. However, real world

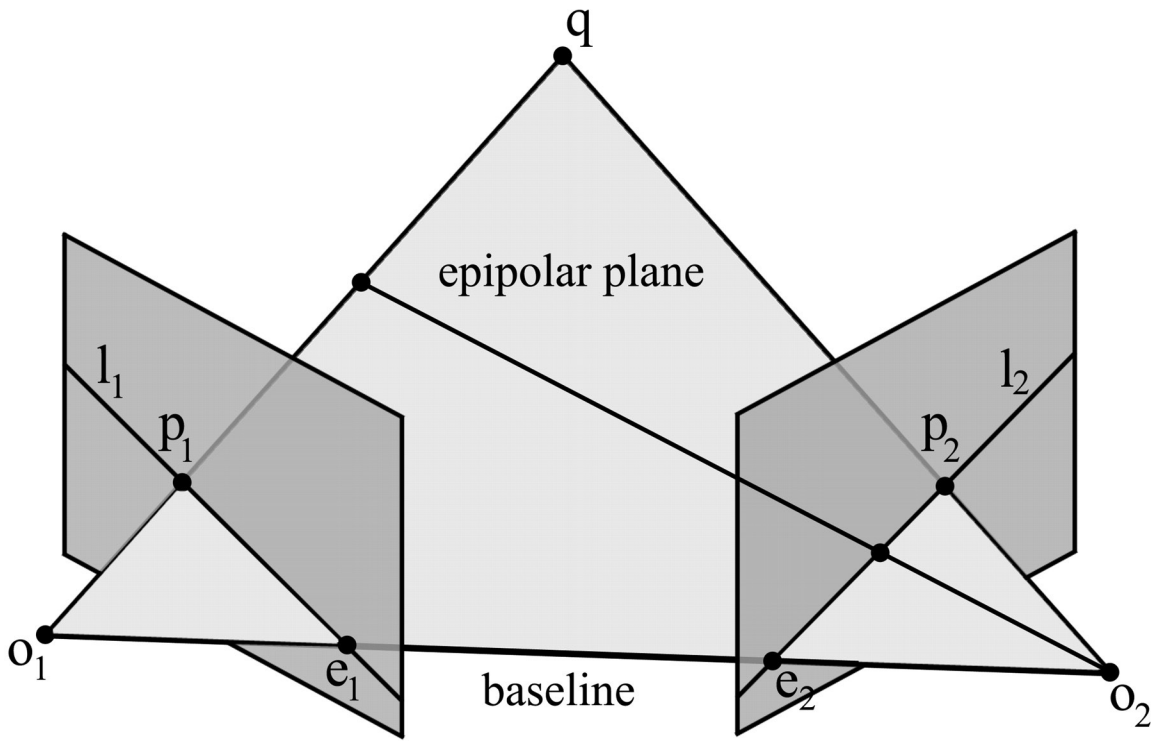


Figure 2. Epipolar Geometry [2]

applications would need to account for parameters such as camera focal length and principal point. Accurate measurement of these intrinsic parameters are necessary for precise position estimation.

2.4 Registration

Registration of point sets, or point matching, refers to the process of aligning two point sets [12]. The sensed point cloud is generated by the stereo vision system. The reference point cloud is generated by a geometrically accurate reference model of the object. The reference model is used as the truth data. Through the process of registration we can estimate the position and orientation of the object.

The iterative closest point (ICP) algorithm is the most common solution for rigid 3D registration problems [13]. ICP works by iterating over the sensed points and searching for a corresponding closest point in the reference model. ICP then calculates

the translation and rotation between the point sets. The sensed point cloud receives this transformation and then ICP restarts beginning at the updated position and orientation. ICP will iterate until a predetermined mean squared error is achieved or until it has reached a maximum number of iterations. A resulting mean squared error close to zero does not guarantee correct registration. ICP must be seeded with the reference model at a starting position and orientation. This seeding can have significant effects on the results of ICP. A bad initial orientation can cause ICP to converge to a local minimum preventing it from finding the global minimum that would produce the optimal registration. There are ways to overcome these local minima such as Go-ICP [14]. The best solution to this problem however is the provide a good initial position and orientation for the reference point cloud. A naive implementation of ICP can be relatively inefficient and for our system we need real time capabilities. Other research has further developed ICP through the implementation of K-D trees to increase efficiency [15]. ICP has been modified and further developed for different purposes focusing on efficiency and accuracy [16, 17, 18, 19, 20]. Our research utilizes a point to point ICP algorithm implemented with K-D trees to improve efficiency necessary for our problem.

2.5 Non-Vision Based Approaches

Global positioning system (GPS) approaches have proven effective for relative navigation with applications in the automated aerial refueling domain. A differential GPS approach was shown to be effective for aerial refueling using the probe and drogue method [21]. Other research has shown that GPS based systems provide centimeter and degree level accuracy for relative navigation formation flight [22, 23, 24]. Thereby providing the relative navigation accuracy necessary for automated aerial refueling. These approaches however, rely on the availability of GPS in contested environments

where GPS accuracy can be degraded or completely denied. Other studies have focused on hardening this approach by combining the capabilities of both GPS and inertial navigation systems (INS) [25]. This research demonstrates that a combined system provides more reliability in the event of one system failure. However, without GPS the ability to accurately estimate position is still degraded and diverges over time. Other research has combined GPS with machine vision through an Extended Kalman Filter (EKF) to produce more robust relative navigation systems [26, 27]. They utilize a monocular vision system installed on the receiver which would require significant aircraft modification.

Another non-vision based approach relies on the use of Light Detection and Ranging (LIDAR) to determine relative positioning between tanker and receiver [28]. This approach is effective but it utilizes an active scanning LIDAR system, potentially degrading the stealth capabilities of military aircraft. The use of LIDAR also requires significant modification to each potential receiver. Such a solution does not scale.

GPS and LIDAR relative navigation systems provide the accuracy needed for automated aerial refueling but concerns over their capabilities in a contested environment persist. The combination of two or more systems have shown to produce more reliable results in the event that one fails but we must harden the system further to appease safety concerns.

2.6 Vision Based Approaches

Computer vision algorithms find use in a wide array of fields. In this section, we discuss a few applications of computer vision, particularly those which are intended to solve the automated aerial refueling problem. We are specifically interested in the application of a stand alone stereo vision system to reduce 6DoF estimation error. Below we describe previous vision-based approaches.

Fravolini's study quantifies the difference in feature matching when using geometrically placed markers and corners with monocular vision. This study also quantified the accuracy and efficiency of several different matching algorithms. Both markers and corner detection produce a relatively high level of accuracy however the maximal clique heuristic presented performed significantly better when combined with corner detection compared to the greedy algorithm [29].

Another one of fravolini's studies quantified the effectiveness of different feature matching methods. Several methods including a Hungarian method, a greedy method and a combined method were compared. All methods were compared based on simulation conducted within a 3D virtual world. Each simulation placed a monocular vision system on the approach receiver and light sources on the underside of the tanker. At close ranges the markers placed on the tanker could be occluded or blocked introducing error spikes into the data. Mitigating the effect of occlusion on features or markers is the basis for Fravolini's research. However, the robust method proposed within this study produced the most stable results never spiking above one meter error even in the event of marker occlusion [30].

Mammarella et al. continued the effort to quanitfy the effectivesness of different point matching algorithms for the UAV refueling problem. They utilized Harris corner detection and Lu, Hager, and Mjolsness pose estimation combined with different point matching algorithms within a 3D virtual world to conduct simulations. Their work compared the mutual nearest point algorithm with the maximum clique detection algorithm. Results show that maximum clique detection provides similar position estimation to mutual nearest point but superior orientation estimation [31].

The use of a monocular vision system placed on the receiver is cost prohibitive and requires significant aircraft modification. Other research has focused on the use of a binocular vision system positioned on the tanker angled down towards the

receiver. Duan, et. al. conducted a hardware-in-the-loop simulation to determine optimal strategies for pose estimation of UAVs in the AAR domain [32]. This study determined that the optimal strategy is to combine LED detection and Gaussian least-squares differential-correction. This combination produces sub-quarter meter error along all axis.

The use of stereo vision for relative pose estimation is not exclusive to the automated aerial refueling problem. Research has shown that the use of stereo vision is an effective solution for autonomous rendezvous and docking for satelites. Simulations conducted using a 1/24 scale Magellan satellite model produced orientation errors of less than 3 degrees and position errors of less than 2% [33].

Previous research at the Air Force Institute of Technology has shown the ability to produce centimeter level and sub-degree 6DoF measurements. All approaches used a stereo vision system installed on the tanker looking down at the receiver. Initial work was simulated in non-real time but proved the ability to conduct an aerial refueling simulation using the stereo vision process [34]. Subsequently, Denby's work focused on the optimization of the stereo vision pipeline to bring it closer to real time. While efficiency was increased the accuracy was degraded to the point where automated aerial refueling would not be safe [35]. Most recently, Parson's research focused on calibration of virtual senors within the 3D virtual world combined with the use of a shelled reference model to imrpove 6DoF measurements and increase real time capabilities. Parson's work presented a real time solution capable of centimeter and degree level accuracy for relative navigation [36].

2.7 Collision Detection

Collision detection is the process of determining if two objects intersect. Our particular application is concerned with ray-plane collision detection or intersection.

Ray-plane collision detection can be utilized to determine if a ray cast through space collides with an object. A naive approach is to iterate over the object's triangles and determine if the ray intersects any of them. This approach can be relatively inefficient if the object consists of a large number of triangles. Another more efficient approach is to utilize an octree for the object of concern. This approach represents the object in a hierarchical 8-ary tree structure [37]. This increases collision detection efficiency as it only iterates down the tree if conditions for the parent nodes are met.

2.8 Shadow Volumes

In computer graphics shadow volumes can be utilized to add shadows to a scene. A shadow volume is defined as the volume of space occluded from a particular light source by an object [38]. In other words any object within this volume will reside in a shadow. An example of a shadow volume can be seen in Figure 3. The light source is a point light source that casts rays equally in all directions. The shadow volume outline or silhouette edge is defined as the edges of the objects that creates the planes of the shadow volume. Silhouette edge detection requires knowledge of the shadow casting object. To determine the edge of the object that creates the silhouette edge we need to know the adjacency information for the triangles that make up the object. We also need to know the normals of these triangles and the position of the light source. With this information a dot product can determine which triangles face, or point toward, the light source. The edge between triangles facing the light source and ones facing away are part of the silhouette edge [39, 40]. We also define the inverse of the shadow volume or the volume of space between the light source and the silhouette edge as the light volume. Initial research on shadow volumes restricted the geometric complexity of the shadow generating object. Aldridge and Woods expanded this work to allow for more complex geometric shape [41].

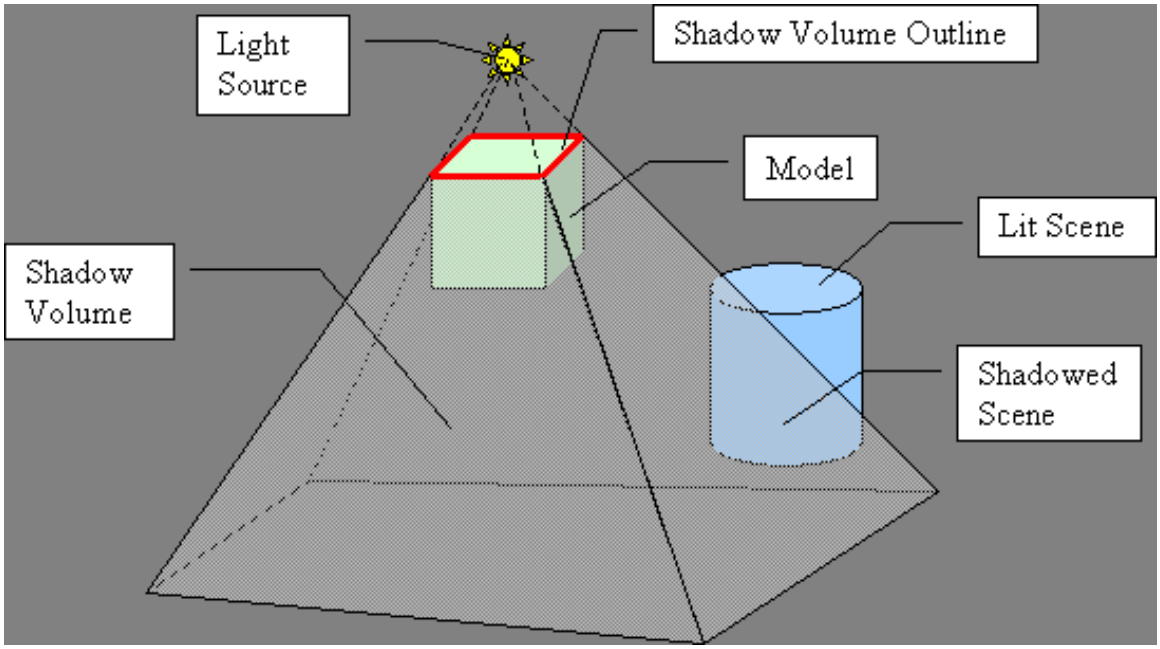


Figure 3. Shadow Volume [3]

Our research is not directly interested in shadows but occlusion can be transformed into a shadow volume problem. When the eyepoint, or pinhole camera, are coincident with the point light source the shadow volume is the volume of space that is occluded or blocked from viewing by the object.

2.9 Point Inclusion

Point inclusion is the process of determining if a point is contained within a volume of space. There are several methods for determining if a point lies within a volume of space. The first is the winding number method [42]. The winding number method has the capability to determine if a point is contained within a concave volume. It accomplishes this by counting the number of times the volume curves around the point of interest, adding one if it wraps counterclockwise and subtracting one if it wraps clockwise. In 2-dimensions this is equivalent to drawing an infinite ray from the point across the object. If the summation is zero then the point is not contained

within the volume of space. The winding number method can be seen in Figure 4.

Another approach exists for simpler convex shapes. This approach calculates the displacement of the triangles that compose the shape or volume. The displacement of each triangle is then compared with the point's displacement. If the point's displacement is less than the triangle's displacement it lies on the inside of the triangle and is within that portion of the shape. Thus, the algorithm iterates over all triangles within an object and if the point lies within all of them the point is in the volume.

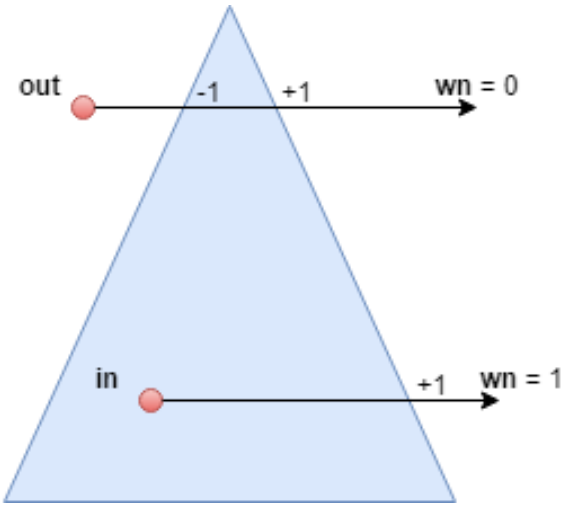


Figure 4. Winding Number Method

III. Methodology

To conduct this research we needed the ability to produce imagery of a flight approach representative of aerial refueling operations. A virtual environment was used to create the simulations allowing us to meet financial and safety limitations. The virtual world also allows for deterministic research that can be conducted in a repeatable manner. To conduct this research in a virtual environment we must create realistic simulations to produce high-fidelity synthetic imagery representative of aerial refueling operations. The Open Computer Vision (OpenCV) libraries contain verified and tested algorithms necessary for our research. We utilize OpenCV's Stereo Block Matching to calculate a point cloud from a stereo pair of images. Our system then uses the iterative closest point (ICP) algorithm to complete point registration of the sensed model onto a reference the model. The 6DoF information necessary for relative navigation is provided by the reference model's position and orientation after registration is complete. ICP performs best when the sensed model and the reference model closely align. Thus, we have two options: we can use a static reference model across the entire flight approach or we can use a dynamic reference model. The dynamic approach is achieved by transforming occluded regions into shadow volumes to dynamically remove occluded points from the reference model. In this way the reference model more closely aligns with sensed points.

3.1 Simulations

All simulations and visualizations for this research use the AFTR Burner engine, an OpenGL based renderer [43]. The AFTR Burner Engine uses models to create realistic synthetic imagery representative of the real world. This work expands the capabilities of the AFTR Burner Engine by introducing adjacency information,

silhouette detection and shadow volumes. Each simulation uses a high-fidelity, geometrically accurate C12 model as the receiver. The C12 was chosen to match the real-world flight tests conducted in September of 2017. The boom model is taken from the KC-10 refueling tanker. The flight path is held constant across all simulations. The flight path is representative of a common approach within the operational limits of the C12 [44]. This flight profile is imported into the virtual world allowing for accurate comparison between sensed and truth data. The virtual sensors used to collect the synthetic imagery maintain a constant baseline of 0.5 meters and mimic specifications of potential real world cameras used for such an operation. The virtual sensors were intrinsically calibrated as described in [36]. All 6DoF results are presented in the coordinate frame system shown in 6. The x component is defined as the red line running from the tail of the receiver to the nose. The y component is defined as the green line running from the right wing of the receiver to the left wing. The z component is defined as the blue line running perpendicular to the xy plane. The orientation components are defined following basic aircraft motion in $roll$, $pitch$, and yaw components. $Roll$ is defined as rotation about the axis running from nose to tail. $Pitch$ is defined as rotation about an axis running from wing to wing. Yaw is defined as rotation about the axis running from top of the aircraft to underside. These axes update as the aircraft moves meaning the Euler angles are defined as rotation about successive coordinate frames. All orientation components follow the right-hand rule.

3.2 Computer Vision Pipeline

A high-level overview of the entire computer vision pipeline can be seen in Figure 7. Pre-processing is required for both camera calibration and the use of shadow volumes. Camera calibration follows the method laid forth by previous research [36]. The shadow volume of the boom is also computed during pre-processing because we

Figure 5. Aerieal Refueling Scene, Rendered in AftrBurner

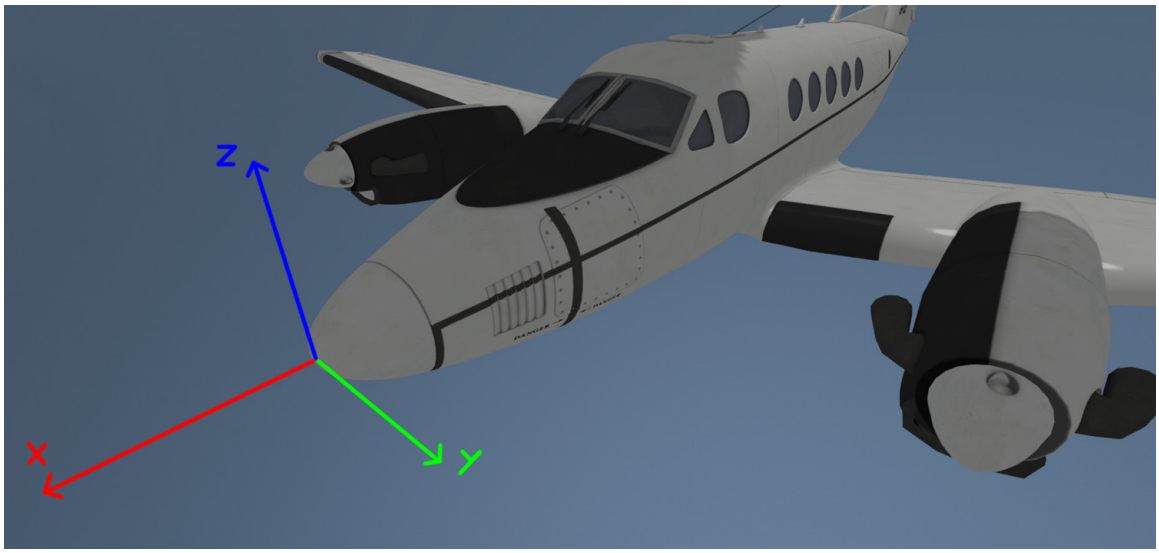


Figure 6. Virtual World Axis

assume a static boom. With adequate knowledge of position and orientation this could be computed in real time during an operation however we are assuming a boom with static position and orientation for this research. The computer vision pipeline outputs a 6DoF estimation for the approaching receiver. The results of the computer vision pipeline with timing metrics allow us to compare performance against other solutions.

3.3 Disparity Map Generation

Once the receiver is traversing along the predetermined flight path log file. The virtual cameras capture the imagery from the perspective of both the left and right camera. These images are passed to OpenCV's Stereo Block Matching function to produce a disparity map. The disparity of a feature is produced by calculating the difference in image coordinates for that specific feature. First the images are rectified, aligning the horizontal line of pixels. This allows the algorithm to calculate the image coordinate difference more efficiently. This is done for all features within the images resulting in the disparity map. OpenCV's Stereo Block Matching function requires two parameters, block size and number of disparities. Block size defines the size of the pixel block that will be treated as a feature. A small block size can result in a significant number of mismatches while a larger block size can result in less granularity. The number of disparities defines the range that OpenCV can assign to pixels. We utilized a block size of 9 and 48 total disparities. These values produced the best results and are consistent with [36] and [45]. The disparity map is then passed through OpenCV's speckle filter to remove outliers. The final disparity map is used in conjunction with the calibration data to create a three dimensional point cloud that is used for point registration. Figure 8 displays a disparity map of the C12 receiver in the refueling envelope with the tanker's boom also in place.

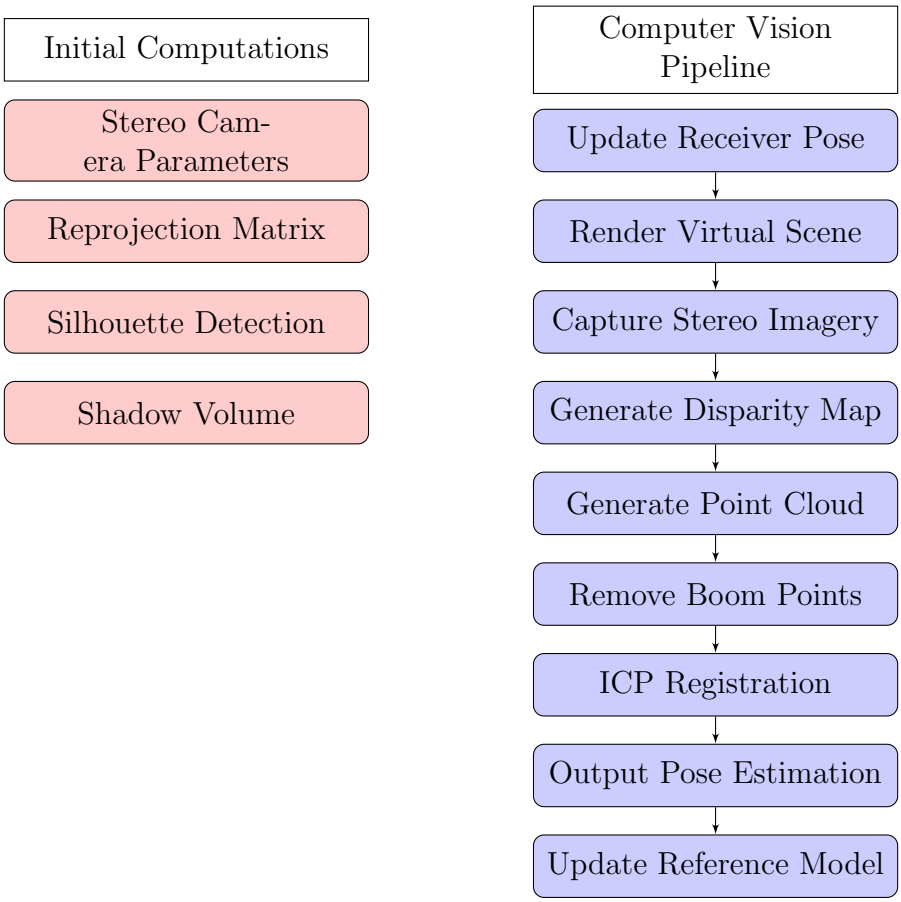


Figure 7. High Level Outline of 3DVW Simulation

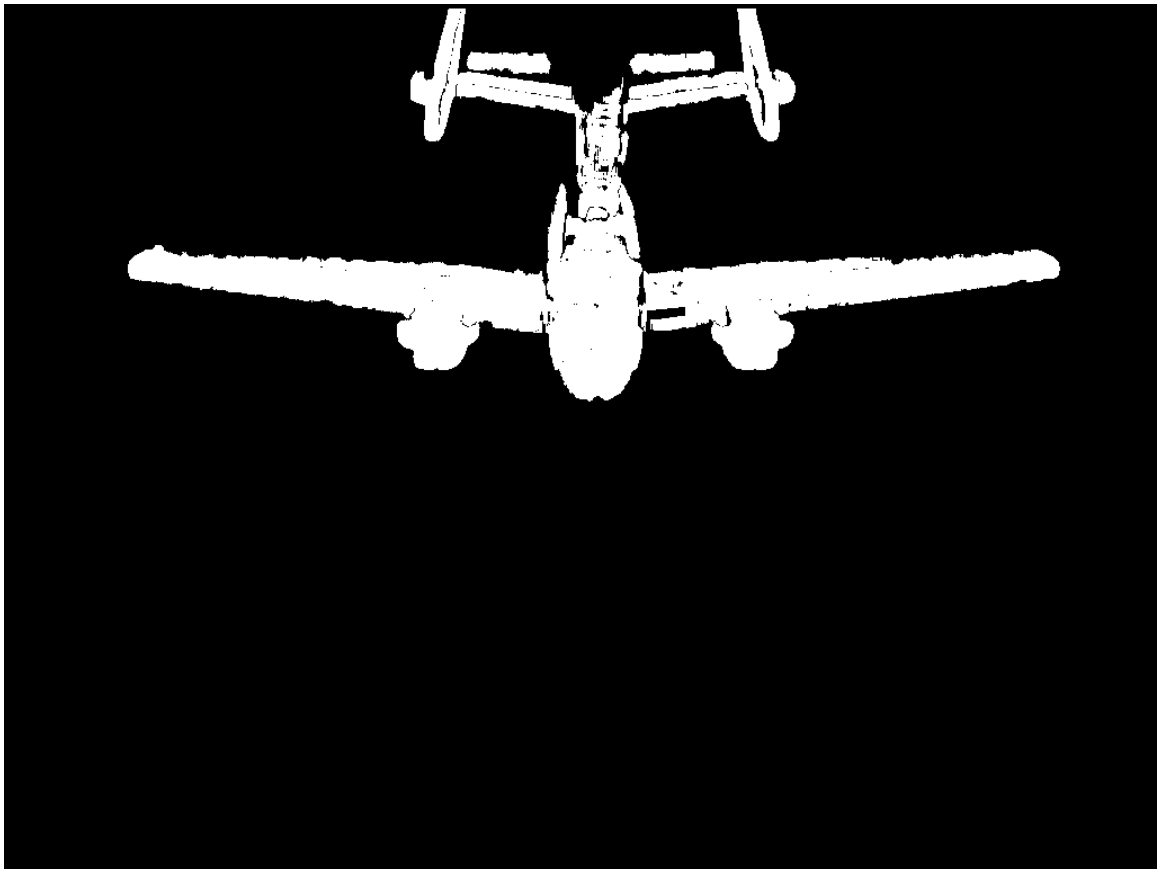


Figure 8. Filtered Receiver Disparity Map

3.4 Point Cloud Generation

These 3D points returned from the disparity map generate a point cloud of sensed points. The reprojection function requires a perspective transformation matrix Q and the pixel coordinates p_x and p_y according to Equation 1.

$$[x\ y\ z\ w]^\top = Q * [p_x\ p_y\ disparity(p_x, p_y)\ 1]^\top \tag{1}$$

The function generates a matrix of 3D points representing both the receiver and the boom. Points greater than 750 meters behind the tanker are filtered out. The system does not accurately sense the receiver at that distance. Thus, removing these points ensures we are only utilizing receiver or tanker generated points and not background. The function generates over 60,000 points at the refueling contact point thus we utilize a uniform decimation removing every eighth point. This decimation is implemented to speed up computation of both ray-collision detection and ICP and was shown in [36] to not have a significant effect on registration accuracy. The points are then transformed into the 3DVW frame and subsequently to the camera's frame through Equation 2 and Equation 3. Where DCM_{tanker} is the tanker direction cosine matrix, DCM_{camera} is the left camera direction cosine matrix and $[x_c\ y_c\ z_c]$ is the left camera position. The resulting point cloud can be seen in Figure 9.

$$[x_v\ y_v\ z_v] = [z\ (-x)\ (-y)] \tag{2}$$

$$[x'\ y'\ z']^\top = (DCM_{tanker} * \left[(DCM_{camera})^\top * [x_v\ y_v\ z_v]^\top\right]) + [x_c\ y_c\ z_c]^\top \tag{3}$$

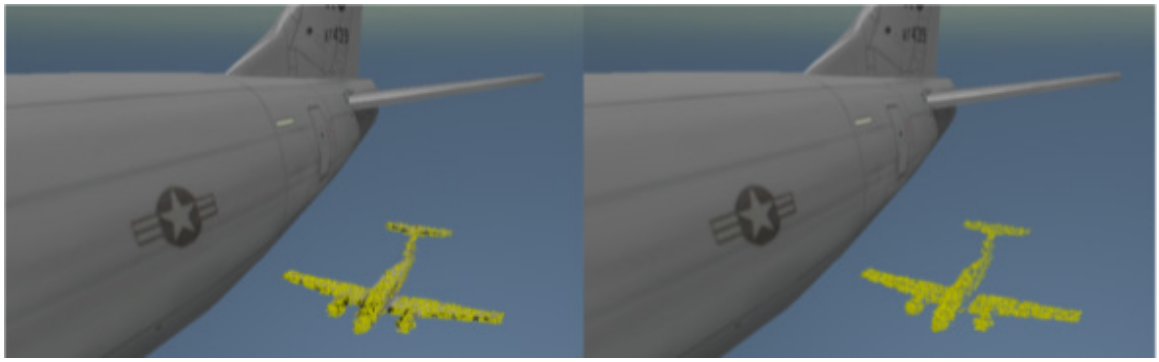


Figure 9. Sensed Point Cloud

3.5 Point Registration

With the sensed point cloud correctly placed in the tanker's left camera's reference frame we can utilize an ICP algorithm to register the reference model of the receiver to the sensed point cloud. The AftrBurner Engine implements a point-to-point ICP algorithm with a modified KD-Tree approach [46]. ICP iterates a maximum of 30 times or until the RMS error is below the previous iterations RMS error. Once completed the position and orientation of the reference model provides our 6DoF estimation corresponding to a single pair of images.

3.6 Static Reference Models

Previous research has shown that the use of a more accurate reference/registration model can not only improve 6DoF estimation accuracy but also improve the performance of ICP. In this previous effort a full reference model of the receiver was taken and modified to create a "shelled model". To produce the shelled model the viewing angle of the reference model was set to match the perspective of the tanker's stereo cameras. The subset containing the visible points of the full reference model were used to create the shelled model. This is helpful because the stereo vision system will never be able to see the underside of the receiver aircraft thus point registration

will encounter errors from these extra points. Using a shelled model gives a more accurate model to match against thereby improving accuracy [36]. The comparison between the full model and the shelled reference model can be seen in Figure 10. Our initial efforts took a similar approach and shelled the model statically before the approach was conducted removing all points not visible due to boom occlusion. The question arose at what point in the approach do we shell the model? This determines which portions of the receiver are occluded. This led us to decide on two specific positions and thus two specific reference models. We classified these models as Wings Only and Wings and Nose, we decided upon these two models as they are the most common occlusions when the receiver is near to the tanker. The Wings And Nose reference model can be seen in Figure 11.Using a more accurate reference model will only improve ICP registration if the sensed point cloud we are matching against closely approximates our reference model. This introduces the next problem of removing boom generated points from our point cloud.

3.7 Ray-Plane Collision Detection

Another obstacle that the boom introduces into the automated aerial refueling problem, besides blocking parts of the receiver, is that stereo block matching generates points that lay on the boom's surface. Without removing these points our ICP algorithm will match the reference model against parts of the boom thus producing incorrect matching and incorrect 6DoF estimation. Thus we needed a way to remove these points given the position and orientation of the boom. One approach is to cast a ray through each sensed point and determine if that ray collides with the tanker's boom. This approach can be seen in Figure 12. Using this approach on the accurate model of the boom does not eliminate all of the points because many of these points lie on the boom's surface, potentially a millimeter or centimeter above the geometric

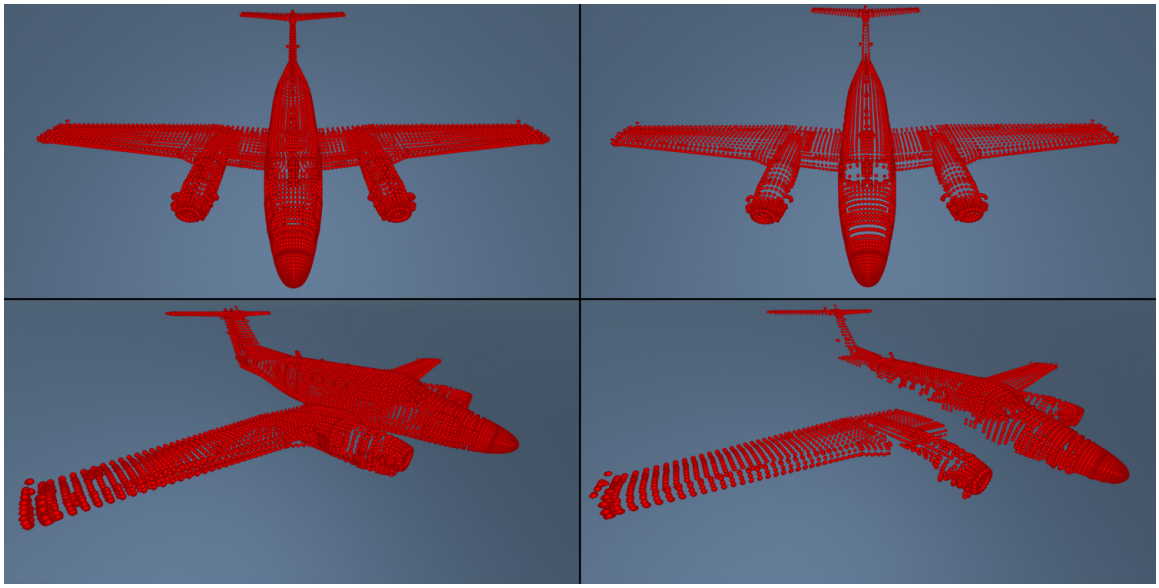
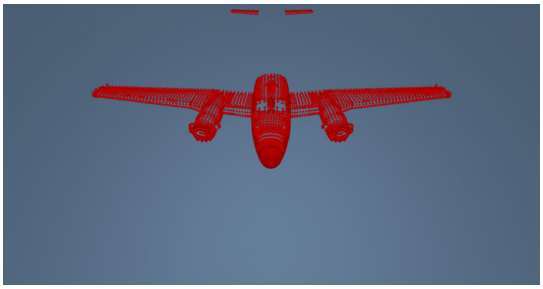
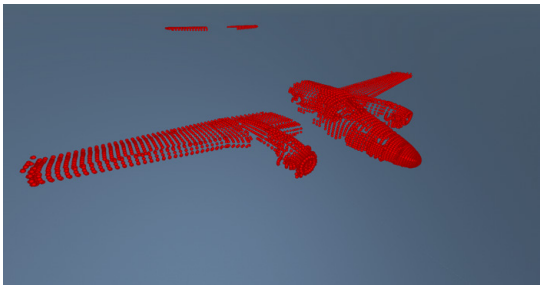


Figure 10. Full (Left) and Shelled (Right) Reference Model



(a) Wings and Nose Top View



(b) Wings and Nose Side View

Figure 11. Wings and Nose Reference Model

surface as a result of corner detection within stereo block matching. The eliminated points are represented as the green points in Figure 13 while the rays casted through the yellow points do not intersect the boom. However, these yellow points clearly lay on the boom; thus we still want to remove them.

We decided to extrude the boom by the least amount possible that allowed us to eliminate 99% of the boom generated points through ray collision detection. Thus we extruded the boom equally in all directions by 10 centimeters. Collision detection against the extruded boom resulted in the elimination of approximately 99% of boom generated points. The resulting collision detection against the extruded boom can be seen in Figure 14. The problem with having a single outlier point resulting from Stereo Block Matching analyzing the boom is that the point will erroneously pull the reference model towards the boom during registration. To improve point registration ICP must have an appropriate seed for both position and orientation. We decided to seed the position of ICP at the average position of all sensed points, thus centering the seed closer to the large mass of points. The orientation of the tanker is used as the orientation portion of ICP's seed. In previous research ICP used the center of the bounding-box for all sensed points as the seed. However, in our work this approach produces a significant increase in error.

There are several problems with ray-collision detection. The O(n) collision detection for approximately 8,000 sensed points against over the 1,600+ faces that create the extruded boom is understandably slow. One common approach for speeding up such an operation is the use of an octree for the object we are doing collision detection against. However, in our instance many of the rays will not collide until late in the octree because of the orientation and position of the boom. Thus, the use of octrees resulted in a minimal speed up for our collision detection algorithm.

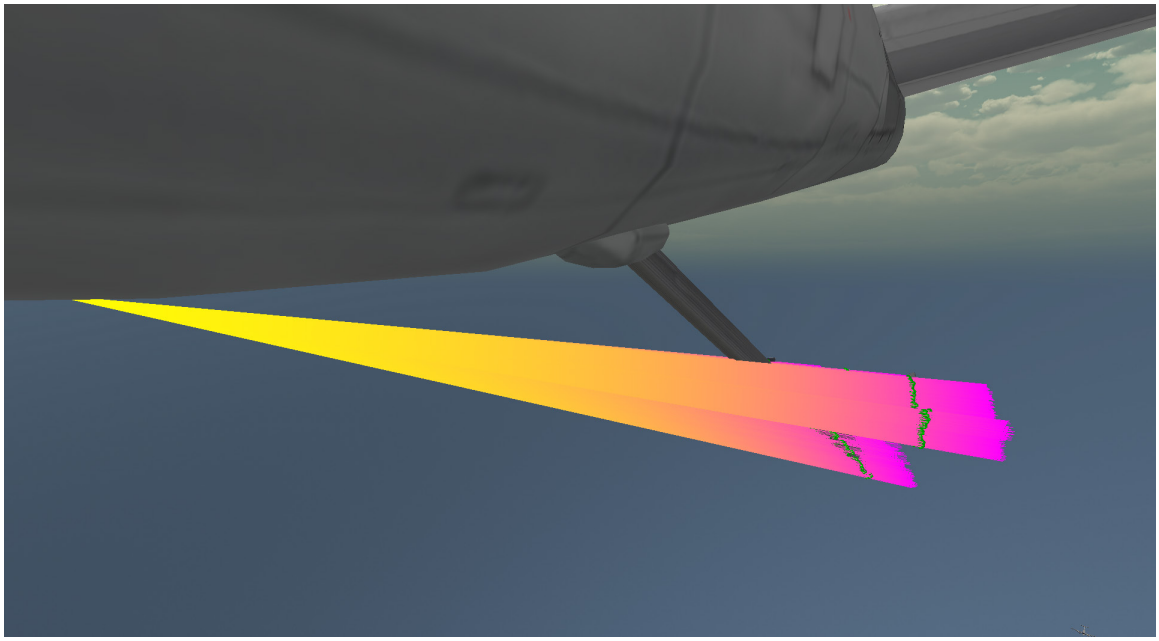


Figure 12. Casted Rays for Collision Detection

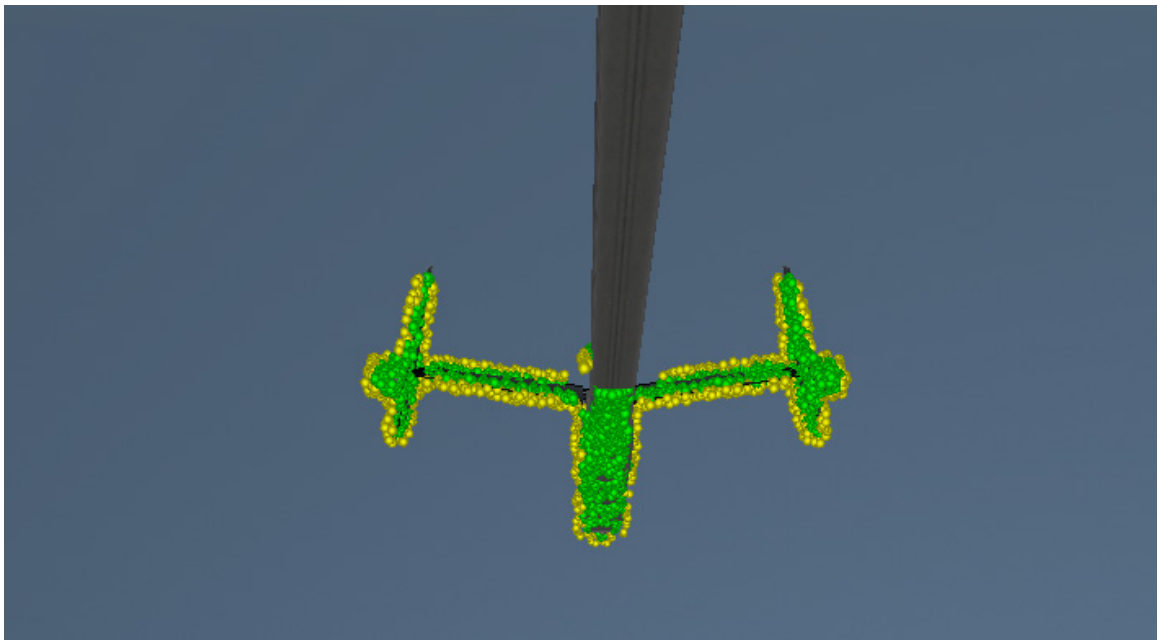


Figure 13. Ray-Collision with Accurate Boom Model

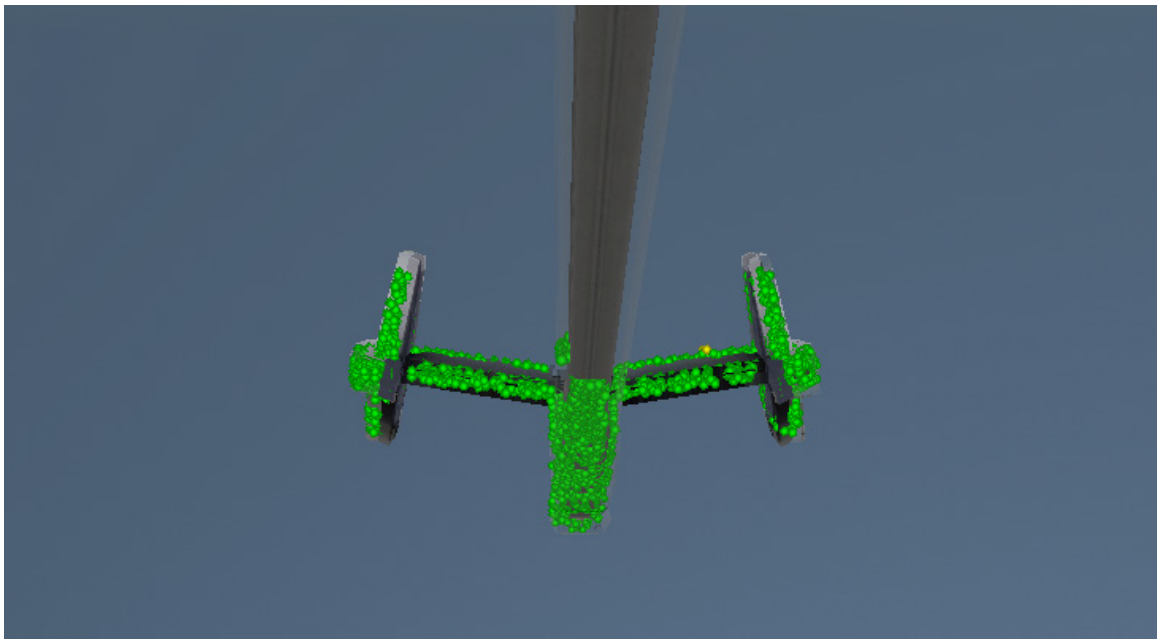


Figure 14. Ray-Collision with Extruded Boom Model

3.8 Shadow Volumes

The other approach is to dynamically create shelled reference models throughout the aircraft's approach. Ideally we would have a reference model exactly mirroring the receiver from the view of the camera. Thus, our reference model would not include any of the pieces of the receiver occluded by the refueling boom. We transform this occlusion problem into a shadow volume problem. A shadow volume is defined as the volume of space occluded from a particular light source by an object [38]. Shadows are not of particular interest in this research however when the light source is coincident with the eyepoint the shadow volume is equivalent to the volume of space not visible to the eyepoint. Our research takes advantage of this fact by utilizing the camera as both the light source and the eyepoint allowing us to determine the exact volume of space not visible by each camera. We utilize this volume of space to dynamically create a shelled reference model that will enhance ICP by more accurately resembling the sensed point cloud.

The dynamic model will be updated every iteration of the pipeline. The model is solely made up of points or vertices and not faces. To update the model we perform point inclusion within the shadow volume of the refeuling boom. To perform point inclusion we need to ensure that our volume is completely convex. The refueling boom of a tanker is not naturally convex and neither was the provided 3D model. Dividing the boom model into twelve convex shapes allowed us to maintain the boom model's high-fidelity geometry while also creating volumes viable for fast point inclusion.

To create an accurate shadow volume we must first determine the silhouette of our object, the tanker's refueling boom. This model is composed of triangles thus our silhouette will be composed of triangle edges. Before we determine what triangle edges are part of the silhouette ring w e must have a model with adjacency information and no duplicate vertices.

Each piece of the boom model is composed of a vertex and index list that define the triangles which create the model. The vertex list is stored as if the model is centered about the world origin. Thus, we will be conducting all of our operations as if the object is centered about the origin as opposed to in world space attached to the tanker. To eliminate duplicate vertices, we perform a simple *O(n)* operation by iterating over all of the vertices and combine any vertices within a certain epsilon of each other into a new a list. This epsilon is defined as .0001 meters in each x, y, z component.

With an accurate vertex and index list we can compute the triangle adjacency information necessary for silhouette detection. We are able to determine if two triangles are adjacent by comparing their vertices, if they share two vertices then they are adjacent. Our adjacency data follows a modified version of the winged-triangle based approach laid out in [47]. Each triangle points at each of its neighbors while also pointing at the three vertices that produce itself.

Once we have the triangle adjacency data for each one of our convex models we can calculate the silhouette ring. To calculate the silhouette ring we must transform the camera's current pose into each convex model's frame (each of the boom's 12 convex models). To position the camera appropriately we perform the transforms in Equation 4 and Equation 5. Where DCM_{boom} is the direction cosine matrix of the boom, $[x_c\ y_c\ z_c]$ is the camera position in world space, $[x_b\ y_b\ z_b]$ is the boom position in world space, DCM_{joint} is the transformation matrix connecting the boom to the tanker in the tanker's frame and DCM_{camera} is the direction cosine matrix of the camera in the world frame.

$$[x'\ y'\ z'] = (DCM_{boom})^{\top} * ([x_c\ y_c\ z_c] - [x_b\ y_b\ z_b]) \tag{4}$$

$$DCM_{camera}{}' = (DCM_{joint})^{\top} * DCM_{camera} \tag{5}$$

With the camera positioned appropriately, in the convex model's frame, we perform a dot product between the vector from the camera to the center of each triangle and the respective triangle's normal. If the triangle is perpendicular to the camera the result of the dot product will be zero and facing away from the camera would be negative while a positive result would mean that the triangle faces towards the camera. We store the result of this dot product alongside the triangle. A triangle edge is part of the silhouette edge if for adjacent triangles one faces towards the camera, a positive dot product, and the other faces away, a negative or zero dot product. We accomplish this by iterating over the adjacency information and checking each adjacent triangle for an opposite signed dot-product. This produces a complete silhouette ring for each boom component.

With the silhouette defined accurately we can produce both the light and shadow

volume of the boom. To create the light volume, which is the inverse of the shadow volume, we create a set of triangles. Each triangle shares the camera's position as a point. The other two vertices of the triangle will be the vertices which make up the sihouette edge. Iterating over all of the edges in the silhoette ring to create these triangles will produce the light volume. To create the shadow volume we take a silhouette edge and extend both vertices along the vector from the camera through each silhouette vertex. These 4 vertices create a plane that will encompass one side of the shadow volume. The dividing surface between the light volume and shadow volume is defined as the triangles on the boom that face the camera. This division is used to cap the shadow volume allowing us to accurately determine what can not be seen from the viewpoint of the camera.

Once the shadow volume is calculated and capped we can divide each plane into two triangles so we can store the three necessary elements for point inclusion. The three elements we need are the normal of each triangle, a point on the surface of each triangle and the displacement of the triangle from the origin. To get this information we use the three points that make up the triangle to calculate the normal of that surface, we then store one of those points. Because we calculated each shadow volume as if the object was centered at the origin if we were to do point inclusion on an object it would require 12 transforms, one for each object. Thus, we decided to transform the normal and point of each plane into world space such that we do not need any additional transformations for point inclusion. Once we have transformed the information we can calculate the displacement shown in Equation 6. Where $[x_p\ y_p\ z_p]$ is a point on the plane and n is the normal of that plane, both of which are in world space. Both the transformed normals and the points on each plane can be seen in Figure 16.

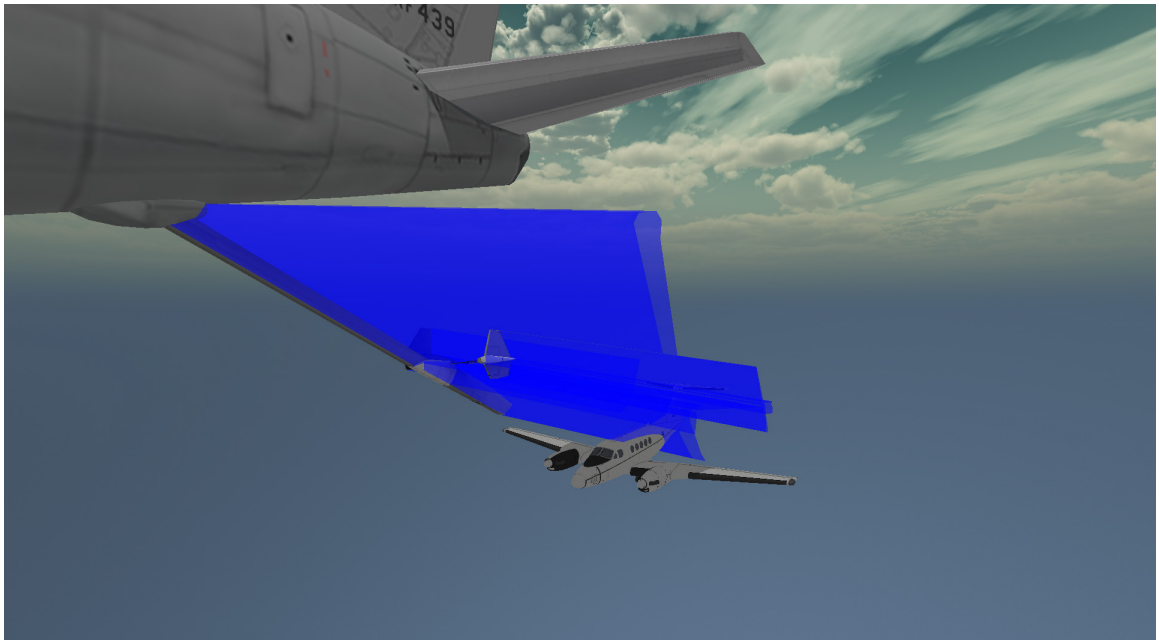


Figure 15. Shadow Volume For Left Camera

$$d = [x_p \ y_p \ z_p] \cdot n \tag{6}$$

To perform point inclusion we simply compare the displacement of any point, calculated by the dot-product between that point and the normal of the plane. If the resulting displacement of the point is less than or equal to the plane's displacement then we know the point lies on the inside of that plane. Thus, we iterate over all of the given planes for a given shadow volume and if the point is within every plane then we know it is within the volume. Doing this from the perspective of both the left and right camera allows us to determine what can be seen by only one camera, by neither camera or by both. In Figure 17 we see point inclusion on the reference model. Red represents the points seen from both cameras, cyan represents what is only seen by the left camera, green represents what is seen by neither camera, and blue represents what is only seen by the right camera. Figure 18 shows the receiver from the perspective of the left camera and we can no longer see the green or blue

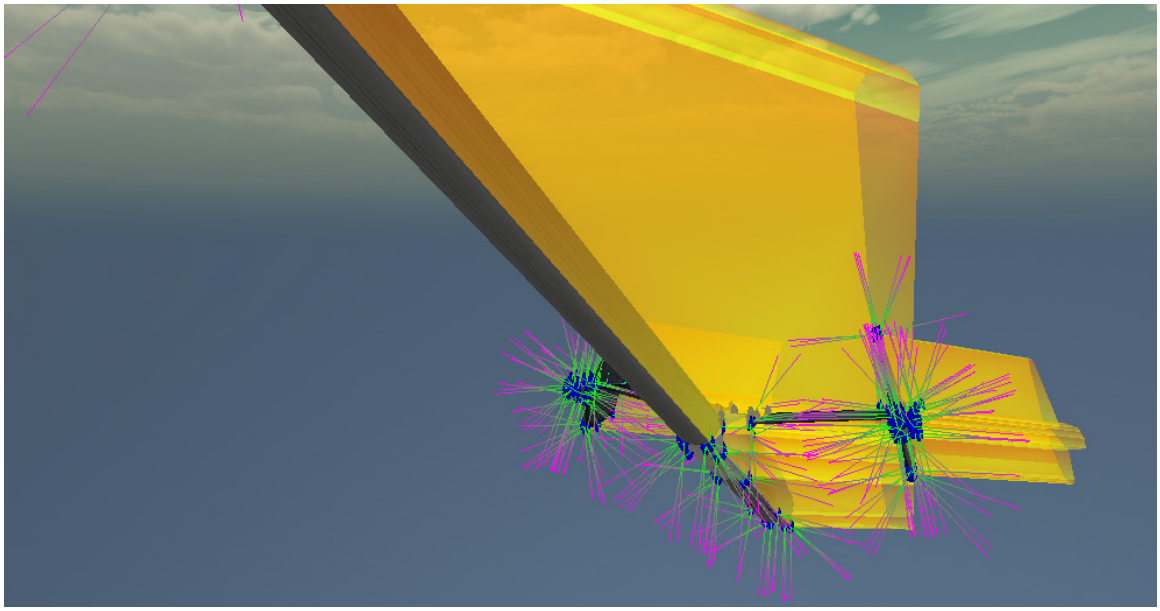


Figure 16. Transformed Normals and Points Required for Displacement

points within the reference model.

With the shadow volumes and point inclusion working correctly we are able to dynamically update the reference model each iteration through the pipeline. This results in a reference model more accurately resembling the sensed 3D point cloud. This can be seen at four different stages of an approach in Figure 19.

3.9 Experimental Design

The stereo computer vision pipeline outlined above produces the data for each experiment. The following subsections outlines each of the three experiments used to produce our results. All experimental results use root-mean-square (RMS) error to compare the accuracy of 6DoF measurements. RMS error is calculated according to equation 7 where i is the position, v_i is the observed estimation. $\hat{v}_i$ is the expected or truth data, and n is the number of samples or iterations through the vision pipeline.

$$RMS\ Error = \sqrt{\frac{\sum_{i=1}^{n}(\hat{v}_i - v_i)^2}{n}} \tag{7}$$

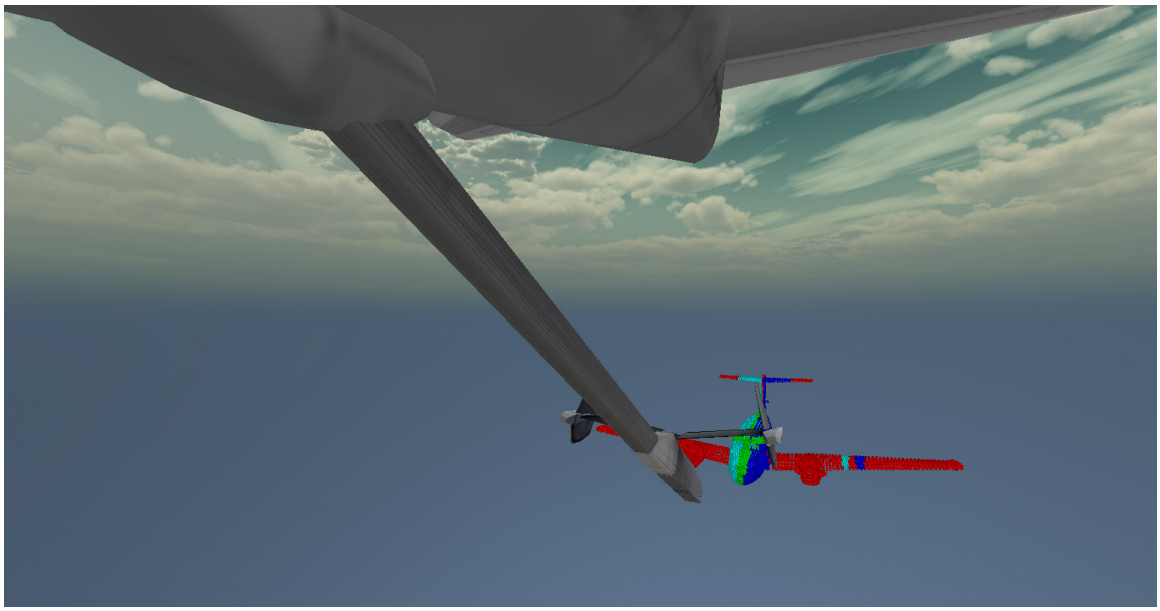


Figure 17. Point Inclusion on Reference Model

Figure 18. Point Inclusion on Reference Model

(a) Dynamic Reference Model 1

(b) Dynamic Reference Model 2

(c) Dynamic Reference Model 3

(d) Dynamic Reference Model 4

Figure 19. Dynamic Reference Model at 4 Stages

Previous research divided the flight path into three regions [36]. The results were divided into these categories because of emphasis on certain estimations at different ranges. The three regions are defined by there distance from the tanker. Region A is defined as 70m to 175m, region B is defined as 38m to 70m, and region C is defined as 32m to 38m. To allow for accurate comparison between solutions we will define our regions in the same fashion. All experiments are conducted with the tanker's boom in view of the vision system.

Experiment 1: Previous Solution With Boom.

Previous research established a baseline of results utilizing the 3DVW however these experiments were conducted with a completely unobstructed view of the receiver. Since, the boom occludes large and varying portions of the receiver on approach another baseline must be established. To establish this baseline we took the previous solution in its best state and conducted the same experiment with the tanker's boom in place. This also allows us to compare the previous solution's results with and without the boom. This naive approach provides no mitigation techniques to handle boom occlusion. To compare the results we calculate the percent difference of the average error magnitude and RMS error for each component of the 6DoF within each region.

Experiment 2: Static Reference Model.

Experiment 2 quantifies the difference in 6DoF estimation between the naive approach and the wings and nose static reference model. In this experiment the boom generated points are eliminated through the ray-collision detection outlined above. To compare the results we calculate the percent difference of the average error magnitude and RMS error for each component of the 6DoF within each region.

Experiment 3: Dynamic Reference Model Through Shadow Volumes.

Experiment 3 quantifies the difference in 6DoF estimation between the naive approach and the dynamic reference model generated through shadow volumes. To compare the results we calculate the percent difference of the average error magnitude and RMS error for each component of the 6DoF within each region.

IV. Results

All experiments were conducted on a Thinkpad P50 laptop with an Intel Xeon E3-1505M v5 processor providing 8 threads to the operating system. The operating system utilized is Windows 10 with 16GB RAM.

4.1 Previous Solution With Boom

The estimations that we will be comparing our work against is the implementation laid out in [36] but with a model of the tanker's boom in the simulation. The results from the previous work, without the boom, can be seen in Figures 20 and 21. Figures 22 and 23 present the error in position and orientation estimation with the boom. These results are compared in Table 1. The results are significantly worse for elements of the 6DoF at every distance from the tanker. The simulations with the boom were also slower with ICP often operating a full 30 iterations before halting.

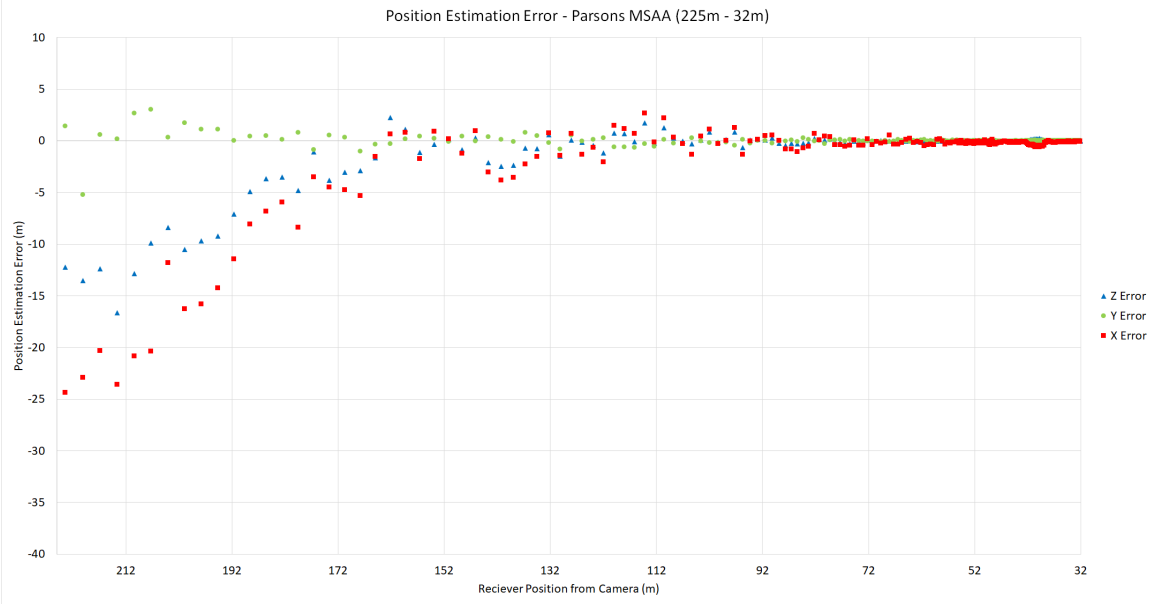


Figure 20. Position Estimation Error for ICP of the Shelled Reference Model with No Boom

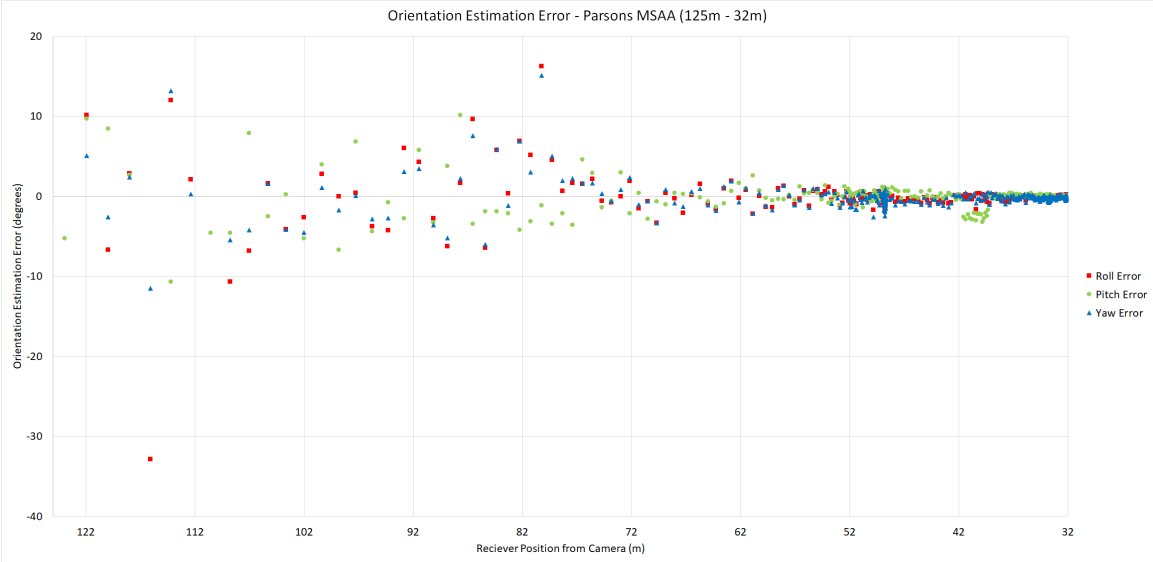


Figure 21. Orientation Estimation Error for ICP of the Shelled Reference Model with No Boom

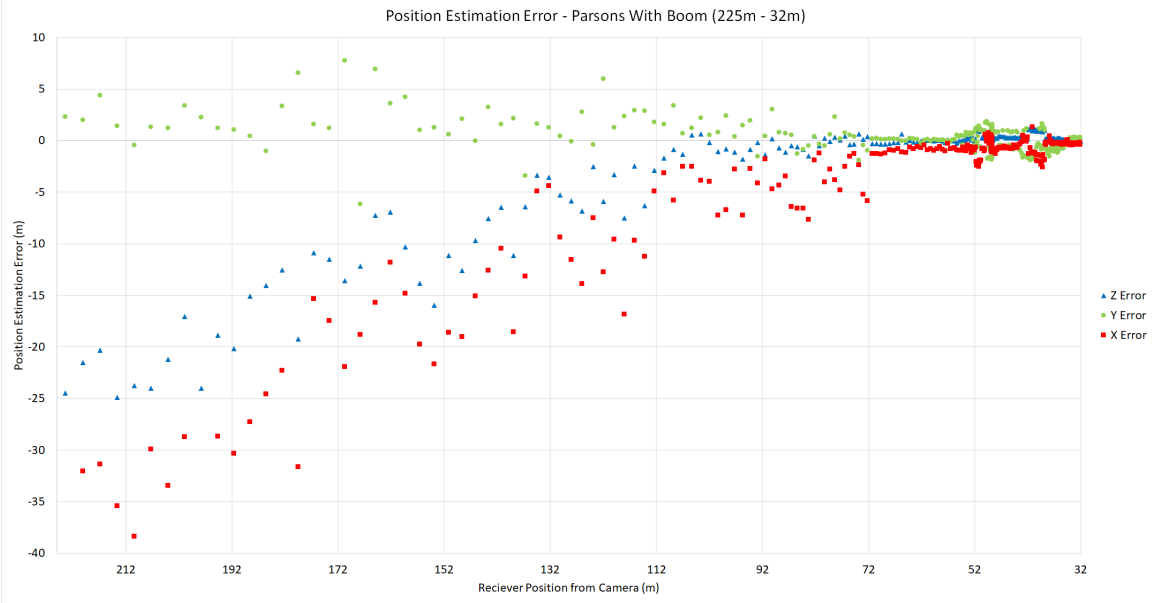


Figure 22. Position Estimation Error for ICP of the Shelled Reference Model with No Boom Mitigation Techniques

It is expected that the results would be worse. The boom generates 1800 points through stereo block matching while at the closest point the receiver generates 6000 points. This means that the tanker's boom is generating at least 23% of the sensed points used in ICP throughout the entire approach. This would result in a signifi-

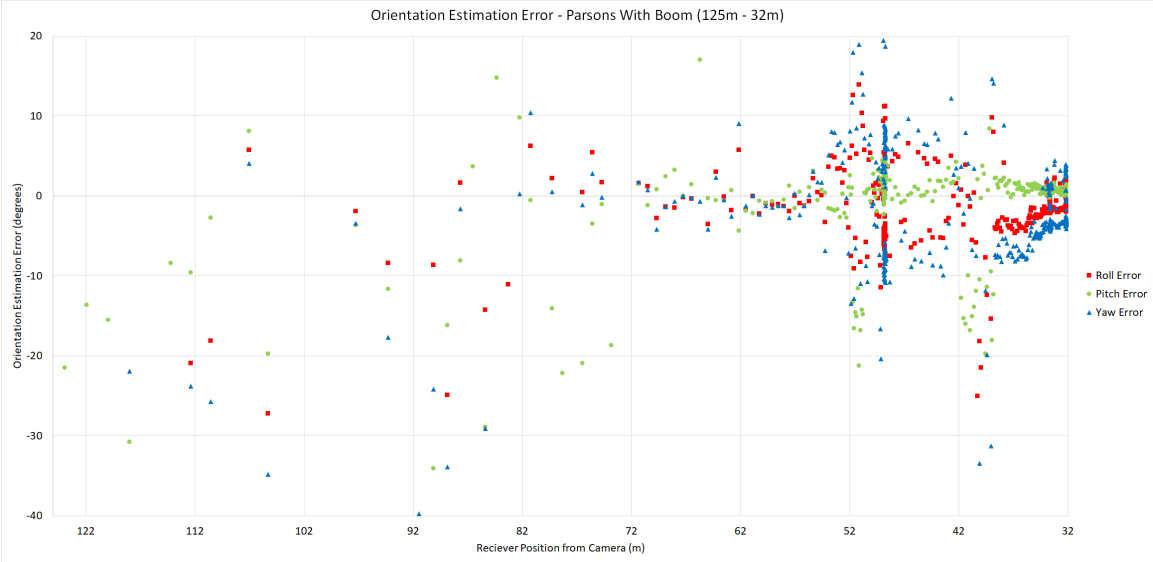


Figure 23. Orientation Estimation Error for ICP of the Shelled Reference Model with No Boom Mitigation Techniques

Table 1. Comparison of Average 6DOF Estimation Error For Approved Flight Path Between Boom and No Boom

		X (m)	Y (m)	Z (m)	R (deg)	P (deg)	Y (deg)
Region A (175-70m)	Boom Average Error Magnitude	8.198	1.725	3.846	69.302	39.404	62.501
	No Boom Average Error Magnitude	1.125	0.236	0.685	6.428	6.966	4.671
	% Difference	-628.91	-630.19	-461.77	-978.07	-465.67	-1238.07
Region B (70-38m)	Boom Average Error Magnitude	0.825	0.880	0.328	6.018	3.463	9.112
	No Boom Average Error Magnitude	0.182	0.030	0.040	0.499	0.614	0.658
	% Difference	-352.80	-2829.18	-724.46	-1105.46	-464.34	-1283.97
Region C (38-32m)	Boom Average Error Magnitude	0.288	0.293	0.032	1.756	0.571	3.556
	No Boom Average Error Magnitude	0.087	0.010	0.010	0.124	0.106	0.183
	% Difference	-230.90	-2931.09	-221.46	-1316.13	-437.37	-1844.74

cant decrease in accuracy. Within 38 meters the least affected component is the z component which still doubles in error.

4.2 Static Reference Model

The combination of a more accurate reference model, consisting of both the wings and nose combined with ray-collision detection to eliminate the boom generated points resulted in significantly better results. The position and orientation estimation error can be seen in Figure 24 and Figure 25 These results are compared with the results

from no boom mitigation techniques in Table 2.

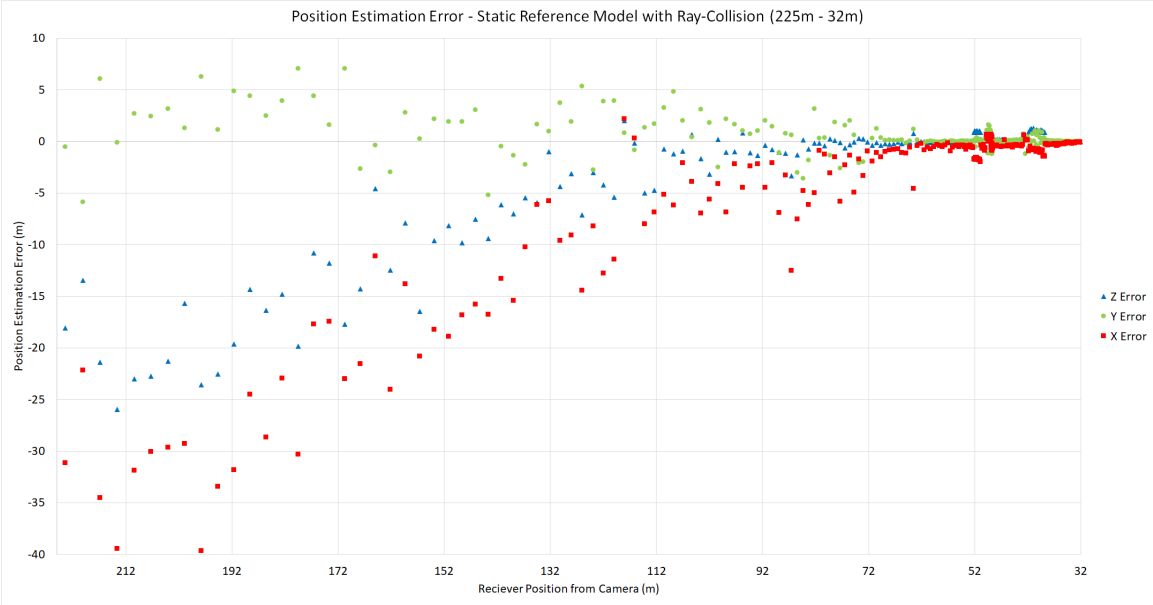


Figure 24. Position Estimation Error for ICP of the Wings and Nose Reference Model with Collision Detection

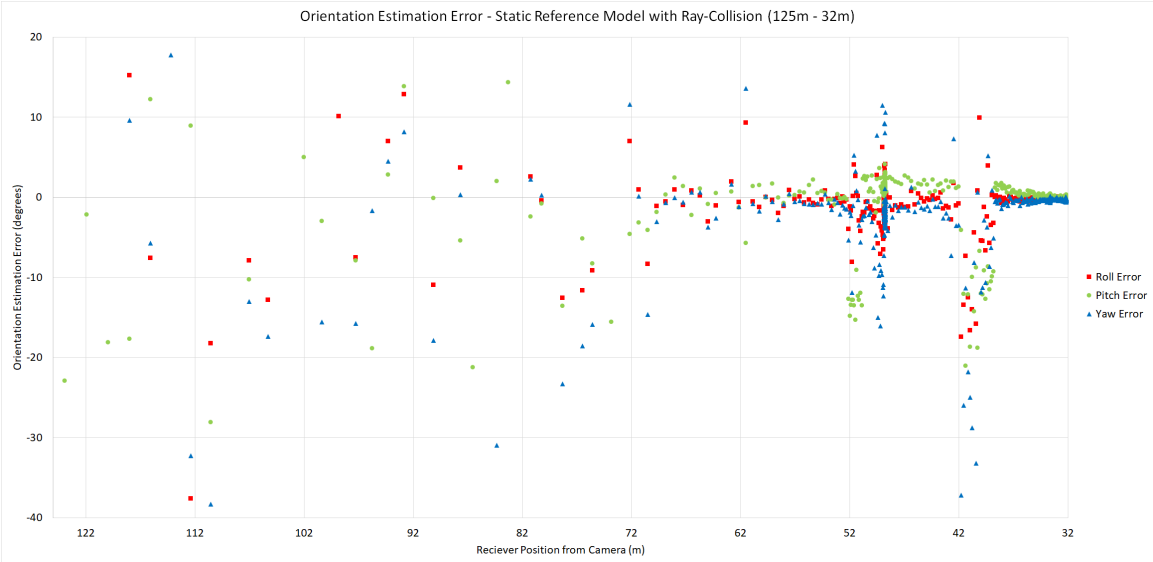


Figure 25. Orientation Estimation Error for ICP of the Wings and Nose Reference Model with Collision Detection

The large error spike seen in orientation when the receiver is approximately 40 meters from the camera can be attributed to the flight approach used in these simulations. The approach follows a NATO certified flight approach however at 40 meters

Table 2. Comparison of Average 6DOF Estimation Error For Approved Flight Path Between Wings And Nose Reference Model and Naive Approach

		X (m)	Y (m)	Z (m)	R (deg)	P (deg)	Y (deg)
Region A (175-70m)	Collision And Wings And Nose Average Error Magnitude	7.975	2.041	3.608	65.199	28.481	53.806
	No Mitigation Techniques Average Error Magnitude	8.198	1.725	3.846	69.302	39.404	62.501
	% Difference	2.72	-18.32	6.19	5.92	27.72	13.91
Region B (70-38m)	Collision And Wings And Nose Average Error Magnitude	0.638	0.264	0.278	3.069	3.697	5.012
	No Mitigation Techniques Average Error Magnitude	0.825	0.880	0.328	6.018	3.463	9.112
	% Difference	22.72	69.98	15.39	49.00	-6.76	45.00
Region C (38-32m)	Collision And Wings And Nose Average Error Magnitude	0.128	0.011	0.007	0.182	0.193	0.314
	No Mitigation Techniques Average Error Magnitude	0.288	0.293	0.032	1.756	0.571	3.556
	% Difference	55.79	96.24	77.17	89.64	66.13	91.17

the receiver's rear wings/horizontal stabilizer go out of the top of the camera's viewing frustum. This error spike while less exaggerated can also bee seen in the results presented in [36] and also in our no mitigation technique results. Overall, the results are significantly improved with the largest increase in accuracy seen for orientation. Reducing our error to less than half a degree within the refueling envelope. The problem with this approach is that at long distances our model does not accurately reflect what the vision system can see. The static model only accurately reflects the model once it is within the refueling envelope.

4.3 Dynamic Reference Model Through Shadow Volumes

Given the performance of the static wings and nose reference model we would expect that a dynamic reference model mirroring the approaching receiver would produce better results. The dynamic reference model is used as input for ICP in the vision pipeline. The position and orientation estimation error of the naive approach can be seen in Figures 22 and 23.The position and orientation estimation error of the dynamic reference model through shadow volumes can be seen in Figures 26 and 27.

The dynamic reference model with collision detection to eliminate boom generated points shows improvement in position and orientation estimation error at all ranges. The largest improvements occur in orientation in Region A and Region C. At close

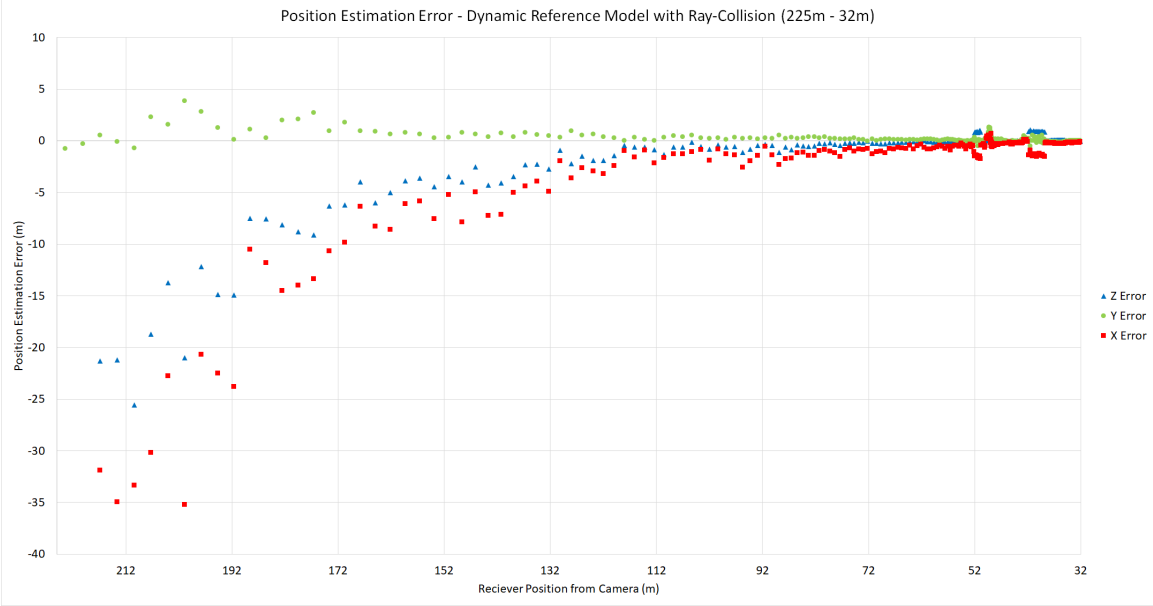


Figure 26. Position Estimation Error for ICP of the Dynamic Reference Model with Collision Detection

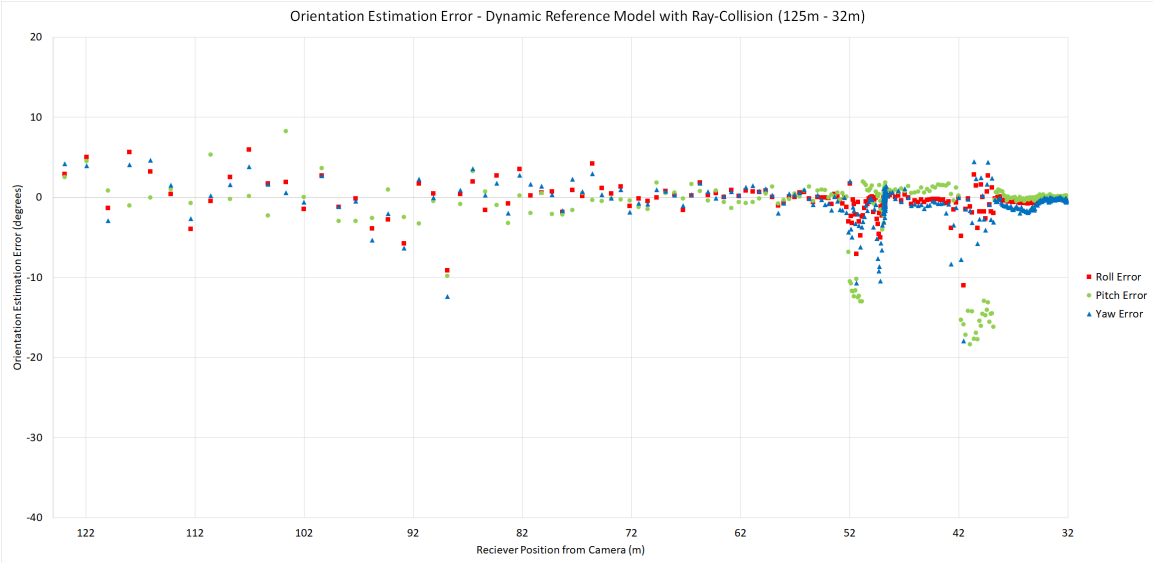


Figure 27. Orientation Estimation Error for ICP of the Dynamic Reference Model with Collision Detection

ranges dynamic model registration reduces error by at least 44%. In Region A the orientation estimation error improves by at least 93% in all components and in Region C the orientation estimation error improves by at least 80% in all components. In Region C the average magnitude of error is approximately 16 centimeters in the X

Table 3. Comparison of Average 6DOF Estimation Error For Approved Flight Path Between Dynamic Reference Model and Naive Approach

		X (m)	Y (m)	Z (m)	R (deg)	P (deg)	Y (deg)
Region A (175-70m)	Dynamic Reference Model Average Error Magnitude	2.943	0.409	1.589	4.205	2.541	3.251
	No Mitigation Techniques Average Error Magnitude	8.198	1.725	3.846	69.302	39.404	62.501
	% Difference	64.10	76.27	58.68	93.93	93.55	94.80
Region B (70-38m)	Dynamic Reference Model Average Error Magnitude	0.551	0.185	0.204	0.894	3.015	1.643
	No Mitigation Techniques Average Error Magnitude	0.825	0.880	0.328	6.018	3.463	9.112
	% Difference	33.26	79.00	37.72	85.14	12.93	81.97
Region C (38-32m)	Dynamic Reference Model Average Error Magnitude	0.160	0.022	0.017	0.195	0.113	0.442
	No Mitigation Techniques Average Error Magnitude	0.288	0.293	0.032	1.756	0.571	3.556
	% Difference	44.64	92.65	47.75	88.90	80.15	87.57

component and 2 centimeters in the Y and Z components.

4.4 RMS Error

RMS error provides expanded insight into the accuracy of our dynamic approach. The RMS errors for ICP across each region are provided in Table 4. At each range, except for Region B's Pitch, our dynamic approach has lower RMS error for 6DoF receiver pose estimation. This further solidifies the performance of our dynamic reference model approach with respect to position and orientation estimation accuracy.

Table 4. RMS Error of 6DoF Estimation For Approved Flight Path Between Dynamic Reference Model and Naive Approach

		X (m)	Y (m)	Z (m)	R (deg)	P (deg)	Y (deg)
Region A (175-70m)	No Mitigation Techniques	10.146	2.403	5.831	92.625	54.040	90.326
	Dynamic Reference Model	3.930	0.510	2.330	6.747	3.669	4.886
Region B (70-38m)	No Mitigation Techniques	0.976	0.999	0.464	14.904	5.686	14.315
	Dynamic Reference Model	0.698	0.298	0.373	1.587	5.894	2.809
Region C (38-32m)	No Mitigation Techniques	0.295	0.356	0.059	1.928	0.697	3.845
	Dynamic Reference Model	0.165	0.042	0.023	0.254	0.150	0.595

4.5 Summary of Contribution

Accurate pose estimation is especially important at close ranges (Region C). Zoomed in plots of both position and orientation estimation error can be seen for the naive approach in Figures 28 and 29. Zoomed in plots of pose estimation error for

the dynamic approach can be seen in Figures 30 and 31. Our approach decreases the estimation error by an average of 73.6% across all 6DoF components within Region C.

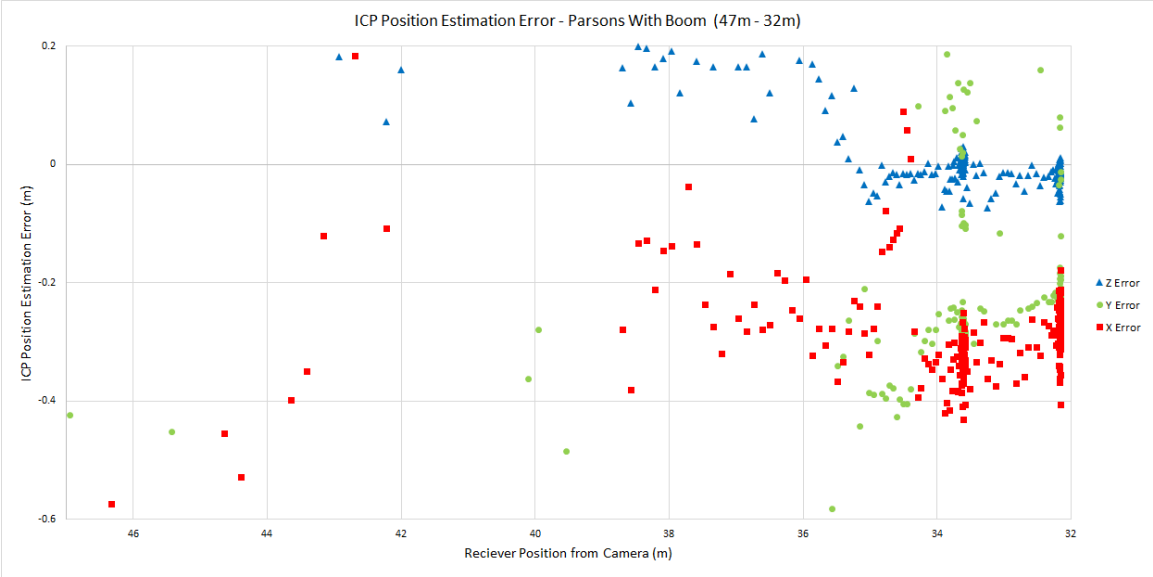


Figure 28. Position Estimation Error for ICP of the Shelled Reference Model with No Boom Mitigation Techniques (Region C)

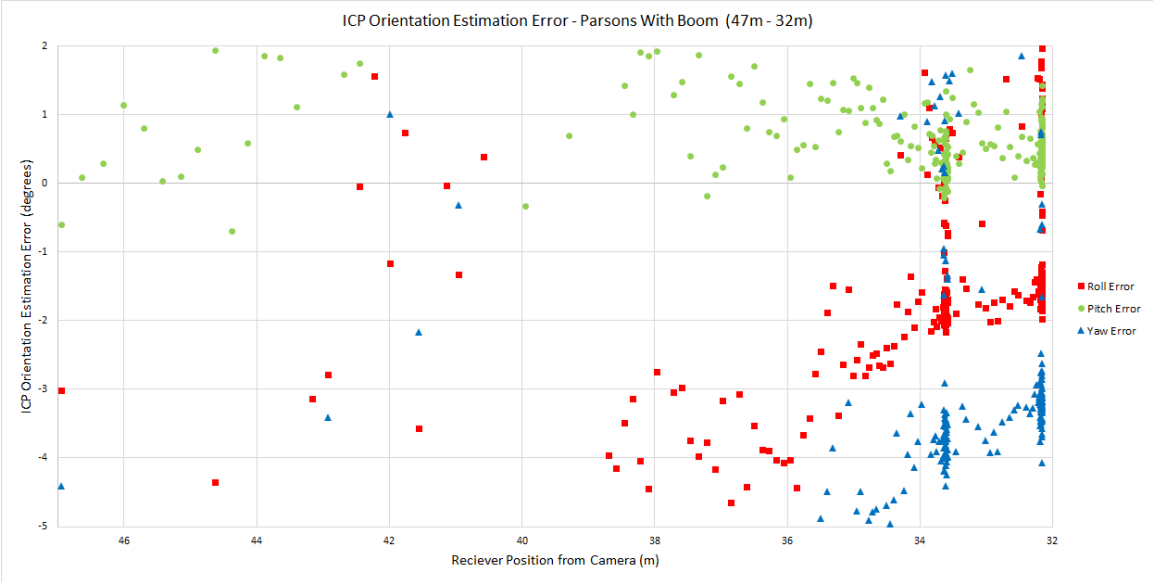


Figure 29. Position Estimation Error for ICP of the Shelled Reference Model with No Boom Mitigation Techniques (Region C)

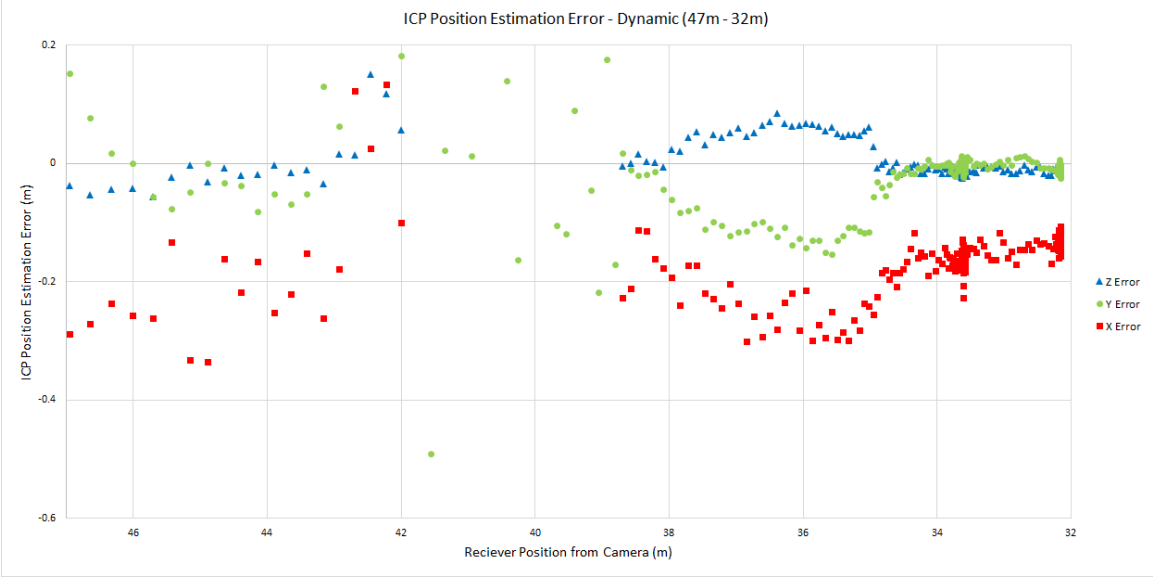


Figure 30. Position Estimation Error for ICP of the Dynamic Reference Model with Collision Detection (Region C)

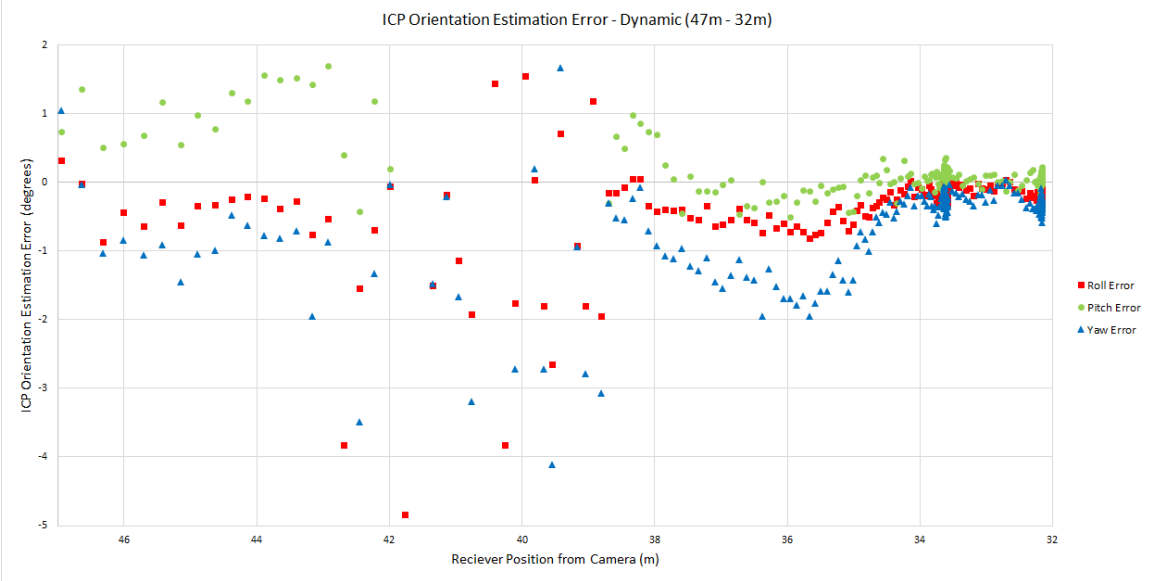


Figure 31. Orientation Estimation Error for ICP of the Dynamic Reference Model with Collision Detection (Region C)

While close ranges are of particular interest, the dynamic reference model produces results that are better throughout the entire approach. The average estimation error for both position and orientation can be seen in Table 5. Pose estimation error decreases by an average of 74.1% across all 6DoF components.

Table 5. Comparison of Average 6DOF Estimation Error For Approved Flight Path Between Dynamic Reference Model and Naive Approach No Regions

		X (m)	Y (m)	Z (m)	R (deg)	P (deg)	Y (deg)
Whole Approach (175-32m)	Dynamic Reference Model Average Error Magnitude	0.659	0.133	0.286	0.967	1.541	1.258
	No Mitigation Techniques Average Error Magnitude	1.443	0.685	0.602	11.491	6.332	12.737
	% Difference	54.31	80.55	52.53	91.585	75.672	90.121

V. Conclusion

5.1 State of AAR

The tanker's boom presents many problems for the automated aerial refueling problem. The effect that occlusion has on relative navigation in [30] was confirmed. This research replaced the monocular system placed on the receiver with the binocular vision placed on the tanker. This research also does not utilize any markers for feature detection but instead relies solely on corner detection in OpenCV's implementation of stereo block matching. It was shown that these negative effects can be mitigated through a more accurate reference model and ray-plane collision detection to eliminate boom generated points. Real-time capabilities of the computer vision pipeline were lost due to the time required for collision detection and the increased time required by our ICP algorithm. The use of a wing and nose reference model combined with collision detection produced the best results for a static model at close ranges.

It was shown that shadow volumes can be utilized to dynamically update the reference model. The use of shadow volumes can determine which parts of the receiver can and can not be seen by either camera. The implementation also operates at a relatively high speed not significantly deterring the potential real time capabilities of our system. Current work is focused on replacing ray-plane collision detection with point inclusion to eliminate boom generated points. This work will utilize point inclusion within the light volumes of each camera to determine which points are generated by the boom. The use of point inclusion will result in a significant speed up, pushing our solution closer to real-time.

This work also contributed to the capabilities of the AftrBurner Engine through the introduction of a triangle/face adjacency information system, silhouette detection and both shadow and light volume creation. Furthermore, the use of a 3D virtual

world to conduct simulations present a deterministic, safe, and efficient way to conduct realistic automated aerial refueling research.

5.2 Future Work

This research makes many assumptions to scope the AAR problem. The tanker's boom is assumed to be static allowing us to calculate the shadow volumes for both the left and right camera before the simulations are executed. With knowledge of the boom's position and orientation the shadow volumes can be updated but would further reduce the efficiency of this pipeline. Aerodynamic forces, such as turbulence, are not present in current simulations and would have effects on both the receiver and the boom. At high rates of speed the aerodynamic forces can cause the boom to flex. This should be approximated and modeled to produce a more realistic simulation.

Propellers found on many UAVs are not present in this research for simplification purposes. Static propellers would occlude portions of both the wings and engines which are the primary matching surfaces for ICP since the boom occludes much of the body. Thus, dynamic propellers with motion blurring techniques must be implemented in the 3DVW to produce more realistic simulations.

This research assumes a pinhole camera model thus geometric lens distortion and blurring of unfocused objects is not present. Introducing camera distortion into the 3DVW would improve the realism of these simulations and allow us to produce imagery more representative of imagery from physical sensors. Camera distortion will also have an effect on the pipeline resulting in less accurate pose estimations. The introduction of camera distortion and the implementation of rectifying it must be completed before the solution can be moved to hardware.

Bibliography

1. Air Refueling Archive. Boeing KC-46A Makes First Contact. Accessed: https://airrefuelingarchive.files.wordpress.com/2016/01/kc-46a-first-contact-23-jan-2016.jpg on 30 January, 2018.

2. Yoram Yekutieli, Rea Mitelman, Binyamin Hochner, and Tamar Flash. Analyzing Octopus Movements Using Three-Dimensional Reconstruction. *Journal of Neurophysiology*, 98(3):1775–1790, 2007.

3. Razvan Surdulescu. CG Shadow Volumes. Accessed: http://archive.gamedev.net/archive/reference/articles/article1990.html on 30 January, 2018.

4. Nicholas J. Seydel. Stereo Vision: A Comparison of Synthetic Imagery vs Real World Imagery for the Automated Aerial Refueling Problem. Master's thesis, Air Force Institute of Technology, 2017.

5. Rob Blackhurst. The Air Force Men Who Fly Drones in Afghanistan by Remote Control. Accessed: http://www.telegraph.co.uk/news/uknews/defence/9552547/The-air-force-men-who-fly-drones-in-Afghanistan-by-remote-control.html on 18 January, 2017.

6. North Atlantic Treaty Organization. *NATO Standard Air-To-Air Refuelling (ATP-56)*. NATO Standardization Agency (NSA), 2013.

7. Simon Prince. *Computer Vision: Models, Learning, and Inference*. Cambridge University Press, 2012.

8. Stuart Russel and Peter Norvig. *Atrificial Intelligence: a Modern Approach*. Pearson, 2009.

9. Muhammad Umair Akhlaq, Umer Izhar, and Umar Shahbaz. Depth Estimation From a Single Camera Image Using Power Fit. In *Robotics and Emerging Allied Technologies in Engineering (iCREATE), 2014 International Conference on*, pages 221–227. IEEE, 2014.

10. David Eigen, Christian Puhrsch, and Rob Fergus. Depth Map Prediction From a Single Image Using a Multi-Scale Deep Network. In *Advances in neural information processing systems*, pages 2366–2374, 2014.

11. Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

12. Richard Szeliski. *Computer Vision: Algorithms and Applications (Texts in Computer Science)*. Springer, 2011.

13. Paul J Besl, Neil D McKay, et al. A Method for Registration of 3-D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

14. Jiaolong Yang, Hongdong Li, and Yunde Jia. Go-ICP: Solving 3d Registration Efficiently and Globally Optimally. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1457–1464, 2013.

15. Zhengyou Zhang. Iterative Point Matching for Registration of Free-Form Curves and Surfaces. *International Journal of Computer Vision*, 13(2):119–152, 1994.

16. Xiao Zhang, Craig Glennie, and Arpan Kusari. LiDAR Using a Weighted Anisotropic Iterative Closest Point Algorithm. 8(7):3338–3346, 2015.

17. Szymon Rusinkiewicz and Marc Levoy. Efficient Variants of the ICP Algorithm. In *3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on*, pages 145–152. IEEE, 2001.

18. D. Chetverikov, D. Svirko, D. Stepanov, and P. Krsek. The Trimmed Iterative Closest Point algorithm. *Object recognition supported by user interaction for service robots*, 3(c):0–3, 2002.

19. Gary K. L. Tam, Zhi-Quan Cheng, Yu-Kun Lai, Frank C. Langbein, Yonghuai Liu, David Marshall, Ralph R. Martin, Xian-Fang Sun, and Paul L. Rosin. Registration of 3D Point Clouds and Meshes: A Survey from Rigid to Nonrigid. *IEEE Transactions on Visualization and Computer Graphics*, 19(7):1199–1217, 2013.

20. Jace Robinson, Matt Piekenbrock, Lee Burt, Scott Nykl, Brian Woolley, and Andrew Terzuoli. *Parallelized Iterative Closest Point for Autonomous Aerial Refueling*, pages 593–602. Springer International Publishing, Cham, 2016.

21. Ryan P Dibley, Michael J Allen, and Nassib Nabaa. Autonomous Airborne Refueling Demonstration: Phase I Flight-Test Results. 2007.

22. Walton Williamson, Jeehong Min, Jason L Speyer, and Jay Farrell. A Comparison of State Space, Range Space, and Carrier Phase Differential GPS/INS Relative Navigation. In *American Control Conference, 2000. Proceedings of the 2000*, volume 4, pages 2932–2938. IEEE, 2000.

23. Mark G Petovello. *Real-Time Integration of a Tactical-Grade IMU and GPS for High-Accuracy Positioning and Navigation*. 2003.

24. Steven M Ross. Formation Flight Control for Aerial Refueling. Technical report, Air Force Institute of Technology Wright-Patterson AFB OH Graduate School of Engineering and Management, 2006.

25. Kevin Liu, Christopher Moore, Robert Buchler, Phil Bruner, Alex Fax, Jacob L Hinchman, Ba T Nguyen, David E Nelson, Fred Ventrone, and Brian R Thorward. Precision Relative Navigation Solution for Autonomous Operations in Close Proximity. In *Position, Location and Navigation Symposium, 2008 IEEE/ION*, pages 1246–1251. IEEE, 2008.

26. Marco Mammarella, Giampiero Campa, Marcello R Napolitano, and Brad Seanor. GPS / MV based Aerial Refueling for UAVs. *AIAA Guidance, Navigation and Control Conference and Exhibit*, (August):1–16, 2008.

27. Marco Mammarella, Giampiero Campa, Marcello R Napolitano, Mario L Fravolini, Yu Gu, and Mario G Perhinschi. Machine Vision/GPS Integration Using EKF for the UAV Aerial Refueling Problem. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(6):791–801, 2008.

28. II Curro and A Joseph. Automated Aerial Refueling Position Estimation Using a Scanning LiDAR. Technical report, Air Force Institute of Technology, 2012.

29. ML Fravolini, V Brunori, A Ficola, M La Cava, and G Campa. Feature Matching Algorithms for Machine Vision Based Autonomous Aerial Refueling. In *Control and Automation, 2006. MED'06. 14th Mediterranean Conference on*, pages 1–8. IEEE, 2006.

30. Mario L Fravolini, Giampiero Campa, and Marcello R Napolitano. Evaluation of Machine Vision Algorithms for Autonomous Aerial Refueling for Unmanned Aerial Vehicles. *Journal of aerospace computing, information, and communication*, 4(9):968–985, 2007.

31. Marco Mammarella, Giampiero Campa, Marcello R Napolitano, and Mario L Fravolini. Comparison of Point Matching Algorithms for the UAV Aerial Refueling Problem. *Machine Vision and Applications*, 21(3):241–251, 2010.

32. Haibin Duan and Qifu Zhang. Visual Measurement in Simulation Environment for Vision-Based UAV Autonomous Aerial Refueling. *IEEE Transactions on Instrumentation and Measurement*, 64(9):2468–2480, 2015.

33. Jed M Kelsey, Jeffrey Byrne, Martin Cosgrove, Sanjeev Seereeram, and Raman K Mehra. Vision-Based Relative Pose Estimation for Autonomous Rendezvous and Docking. In *Aerospace Conference, 2006 IEEE*, pages 20–pp. IEEE, 2006.

34. Kyle P. Werner. Precision Relative Positioning for Automated Aerial Refueling from a Stereo Imaging System. Master's thesis, Air Force Institute of Technology, 2015.

35. Bradley D. Denby. Towards Automated Aerial Refueling Real Time Position Estimation With Stereo Vision. Master's thesis, Air Force Institute of Technology, 2016.

36. Christopher A Parsons. Improving Automated Aerial Refueling Stereo Vision Pose Estimation Using a Shelled Reference Model. Master's thesis, Air Force Institute of Technology, 2017.

37. Donald Meagher. Geometric Modeling Using Octree Encoding. *Computer graphics and image processing*, 19(2):129–147, 1982.

38. Franklin C Crow. Shadow Algorithms for Computer Graphics. In *ACM SIGGRAPH Computer Graphics*, volume 11, pages 242–248. ACM, 1977.

39. Mark Harris, Shubhabrata Sengupta, and John D Owens. GPU Gems 3. 2007.

40. ETAY Meiri. Modern OpenGL Tutorials. *WWW document, http://ogldev. atspace. org*, 2014.

41. Graham Aldridge and Eric Woods. Robust, Geometry-Independent Shadow Volumes. In *Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, pages 250–253. ACM, 2004.

42. Eric Haines. Point in Polygon Strategies. *Graphics gems IV*, 994:24–26, 1994.

43. Scott Nykl, Chad Mourning, Mitchell Leitch, David Chelberg, Teresa Franklin, and Chang Liu. An Overview of the STEAMiE Educational Game Engine. In *Frontiers in Education Conference, 2008. FIE 2008. 38th Annual*, pages F3B–21. IEEE, 2008.

44. Bluemax6. Accessed: https://www.dsiac.org/resources/models_and_tools/bluemax6 on 27 Novembers, 2017.

45. Daniel T Johnson, Scott L Nykl, and John F Raquet. Combining Stereo Vision and Inertial Navigation for Automated Aerial Refueling. *Journal of Guidance, Control, and Dynamics*, 2017.

46. Jace Robinson, Matt Piekenbrock, Lee Burchett, Scott Nykl, Brian Woolley, and Andrew Terzuoli. Parallelized Iterative Closest Point for Autonomous Aerial Refueling. In *International Symposium on Visual Computing*, pages 593–602. Springer, 2016.

47. Christer Ericson. *Real-Time Collision Detection*. CRC Press, 2004.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
23–03–2018	Master's Thesis	Sept 2016 — Mar 2018

4. TITLE AND SUBTITLE

Mitigating the Effects of Boom Occlusion on Automated Aerial Refueling Through Shadow Volumes

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Paulson, Zachary C., 2nd LT, USAF

5d. PROJECT NUMBER

16G189

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-MS-18-M-051

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Ba T Nguyen
Aerospace Systems Directorate, Air Force Research Laboratory
2210 8TH ST
WPAFB OH 45433-7765
(937) 938-7765
Email: ba.nguyen@us.af.mil

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/RQ

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

In flight refueling of Unmanned Aerial Vehicles (UAVs) is critical to the United State's Air Force. However, the large communication latency between an operator and his/her remote UAV makes aerial refueling unsafe. This latency may be overcome by a tanker-based vision system that observes and computes an approaching receiver's relative pose. Unfortunately, the boom – an arm responsible for physically pumping fuel into the receiver – occludes large portions of the receiver. The vision system must be able to compensate for the boom's occlusion of the receiver. Our algorithm dynamically compensates for occluded receiver geometry by transforming the occluded areas into shadow volumes. These shadow volumes are then used to cull hidden geometry that is traditionally consumed, in error, by the vision processing pipeline.

15. SUBJECT TERMS

Automated Aerial Refueling, Virtual World, Computer Vision, Position Estimation, Relative Navigation, Shadow Volumes

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Scott Nykl, PhD, AFIT/ENG
U	U	U	U	68	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636, x4395; scott.nykl@afit.edu