Naval Information Warfare Center

PACIFIC

TECHNICAL REPORT TR-3248
October 2021

Radar Navigation Performance and Final Report

Eric Bozeman

Minhdao Nguyen

Mohammad Alam

NIWC Pacific

Naval Information Warfare Center Pacific (NIWC Pacific)
San Diego, CA 92152-5001

This page is intentionally blank.

TECHNICAL REPORT TR-3248
October 2021

Radar Navigation Performance and Final Report

Eric Bozeman

Minhdao Nguyen

Mohammad Alam

NIWC Pacific

Administrative Notes:
This report was approved through the Release of Scientific and Technical Information (RSTI) process in January 2021 and formally published in the Defense Technical Information Center (DTIC) in October.

NIWC Pacific
San Diego, CA 92152-5001

NIWC Pacific
San Diego, California 92152-5001

A. D. Gainer, CAPT, USN
Commanding Officer

W. R. Bonwit
Executive Director

ADMINISTRATIVE INFORMATION

The work described in this report was performed by the Non-Linear Dynamics & Materials Branch of the Basic & Applied Research Division, Naval Information Warfare Center Pacific (NIWC Pacific), San Diego, CA. The work was sponsored by Dr. Tommy Willis of ONR Code 312, Precision Navigation and Timekeeping, and the work is performed by personnel of Naval Information Warfare Center (NIWC) Pacific.

Released by

Ayax Ramirez Division Head
Basic & Applied Research Division

Under authority of

Carly Jackson, Head
Cyber / S&T Department

Editor: RJP

EXECUTIVE SUMMARY

This report compiles the work done throughout the course of the Radar Navigation (RadarNav) project. RadarNav was funded by Dr. Tommy Willis of ONR Code 312, Precision Navigation and Timekeeping. The work was carried out by Naval Information Warfare Center (NIWC) Pacific; formally Space and Naval Warfare Systems Center (SPAWAR) Pacific. RadarNav is a navigation solution for surface vessels in littoral areas without access to GNSS (Global Navigation Satellite Systems). This report includes a detailed description of the RadarNav system as well as the results of several demonstrations and data collection events. Additionally, there are recommendations for the future use and implementation of RadarNav, which includes possible improvements to the system.

The results of this effort show that the RadarNav system is capable of providing a navigation solution for surface vessels in littoral areas when GNSS is not available. RadarNav accomplishes this with no additional hardware, and only requires a computer with adequate processing power to run the software. RadarNav's limiting factors are related to the quality of available maps and the performance of the radar system in use. The majority of Navy vessels should have access to high quality maps and high-performance radar.

This page is intentionally blank.

ACKNOWLEDGEMENTS

The RadarNav effort has had several project managers and engineers working on it over the years. This section is meant to acknowledge past members of the project who were not directly involved in this report, but whose work was crucial to the project. These are former project members from Naval Information Warfare Center (NIWC) Pacific (formally Space and Naval Warfare Systems Center Pacific):

- Jacoby Larson
- Chris Barngrover
- Amin Rahimi
- Justin Gorgen
- Jim Edwards
- Garret Catron

In addition to the former project members, this effort would not have been possible without the efforts of the volunteers that collected the data presented in this report. A special thanks goes out to:

- Trevor Young and the crew of the RV Kilo Moana from the University of Hawaii
- Leif Manley from Naval Information Warfare Center (NIWC) Atlantic
- Eric Flament from Naval Information Warfare Center (NIWC) Atlantic
- Michael Ferguson from Naval Information Warfare Center (NIWC) Pacific
- Jun Fajardo from Naval Information Warfare Center (NIWC) Pacific
- Niclas Svensson from the Swedish Defence Materiel Administration (FMV)
- Jon Alberts at University-National Oceanographic Laboratory System (UNOLS)

This page is intentionally blank.

CONTENTS

FIGURES

TABLES

1 BACKGROUND

1.1 PURPOSE

The goal of this research effort is to provide a radar-based navigation solution in littoral areas that does not rely on GNSS (Global Navigation Satellite Systems). The radar navigation system used for this effort was designed and built at Naval Information Warfare Center (NIWC) Pacific; previously Space and Naval Warfare Systems Center (SPAWAR). For the remainder of this document, this system will be referred to as RadarNav. RadarNav takes advantage of the radar systems that already exist on most commercial and government vessels. GNSS has proven to be a valuable and highly accurate PNT (Positioning, Navigation and Timing) solution for many military applications. However, it is also very susceptible to jamming and interference, and it is not readily available under water or indoors. Nevertheless, the number of systems that rely on GNSS continues to increase, and now that list includes many critical infrastructure systems. In order to decrease the military's dependence on GNSS, a great deal of research has been put into GNSS-denied PNT alternatives. RadarNav is one such alternative. Radar is already used to manually navigate in littoral areas near a coastline, around low-lying objects that may be difficult to see onboard a ship, or in times of limited visibility (usually due to weather). The purpose of this research effort and report is to automate the process of determining position based on the returns of the radar system. Additionally, the RadarNav system is capable of providing error estimates for its position solutions, as well as predicted positional accuracy based on geography and radar capabilities. The latter can be used for mission planning, and helps lead the way for using RadarNav in autonomous environments.

1.2 THEORY OF OPERATION

RadarNav is a type of map matching navigation system that uses the vessel's navigation radar as it's sensor. Other map matching navigation systems use LIDAR or sonar for their sensor. Regardless of the type, all of these systems use the sensor to collect range measurements between the sensor and a stationary object like coastal geographic features or the sea floor. These range measurements are compared to existing maps, and determine the sensor's position by matching features found in the range measurements to those on the map. All of these systems share several common limitations. The most prominent limitations are the quality of the maps being used for comparison, and the quality of the range measurements. Coastal geographic features are typically easier to work with due to the abundance of surface maps, and the lack of sea floor maps. When it comes to the sensor, the major limitation is that the stationary objects can easily be too far away from the surface vessel for the sensor to "see". Therefore, map matching navigation systems on surface vessels are usually intended for littoral areas limited by either the distance to shore, or the depth of the water. RadarNav has several advantages over its map matching competitors. Many navigation radars operate in the X band (8 – 12 GHz) of the electromagnetic spectrum. With these microwave frequencies the radar is able to work through clouds and fog. It can also provide range measurements from tens of miles away from shore. One of the biggest advantages however, is that most of the Navy's surface vessels already have a radar system installed. Figure 1 shows three different types of radar systems installed on U.S. Navy ships of different size/class.

AN/SPS-10 (Raytheon)	AN/SPS-73 (Furuno)	DRS4D-NXT (Furuno)
C-Band Surface Search Radar	X-Band Surface Search Radar	I-Band Naval Navigation Radar
Used on crusiers	Used on destroyers & carriers	Used on patrol & riverine boats

Figure 1. Three types of radar installed on Navy ships of various sizes and classes.

RadarNav uses standard surface maps to create simulated radar images that are a close approximation of what the true radar data would look like. A simulated radar image is created for each possible location of the vessel. These simulated radar images are then compared to the true image from the radar system. Each time a true radar image is compared to the set of simulated images, the number of possible locations is reduced, until only a few locations remain. The location with the highest probability (or best match between simulated and true radar images) becomes the RadarNav output, and provides the user their position. With no initial position information provided on startup, RadarNav would create an initial set of location points that covers the entire world (at least all large bodies of water). This is not practical, so RadarNav is always provided a general area to bound its initial solutions. Fortunately, in a real-life scenario, the crew of a naval vessel would always have a good enough idea of where they are to provide this bounding information. By the time RadarNav is started up, the last valid GPS position should still be accurate enough to reasonably bound the initial search area (i.e. limited to tens of square miles, not the Pacific Ocean). For the purposes of this research, RadarNav was initially restricted to a single map section, whose geographical size depends on the location and maps being used. In addition to providing a position solution, RadarNav also provides an error estimate based on the available data and how well the true and simulated radar images match. This is known as the estimated error, and it allows the user to assess the accuracy and usefulness of RadarNav in real time.

Additionally, RadarNav has an availability analysis tool that can be run prior to deployment. Given a position, or set of positions, the Availability Analyzer provides the best possible accuracy at each position. This accounts for the coastal geography and quality of the maps being used, as well as the power, quality and range of the radar system to be used. This availability analysis tool would be useful for mission planning in areas where RadarNav might be essential. Users would have the ability to plot a course that ensures RadarNav would always be as accurate as possible.

2 HARDWARE

In an operational scenario, RadarNav would be provided with digital data from the vessel's radar system(s), and would run in real time. Ideally, there would be some sensor fusion managing the position solutions from GPS, RadarNav, and any other navigation system on board. For the purposes of this research effort, RadarNav's ability to operate in this capacity was proven during the real-time demonstrations. However, this required a boat with a decent radar system that the project was able to control and alter. While this was feasible in and around the San Diego area, similar testing in other locations would have required either shipping our boat to that location, chartering a boat with an acceptable radar, or chartering a boat that allowed our radar to be installed. All of these options would have significantly increased cost and logistics complexity. Therefore, it was decided that alternate locations would be tested using post-processed data collected from research vessels that volunteered to collect radar data for this effort.

2.1 REAL-TIME DEMONSTRATION HARDWARE

For the real-time demonstration and testing, the boat that was available was a 22' Boston Whaler. The radar was a Furuno FAR-2117 with a 4' XN12AF antenna; which was oversized for the vessel. Figure 2shows this antenna.

Figure 2. XN12AF radar antenna on RadarNav boat.

The radar was mounted on a telescoping carbon fiber pole just aft of the Bimini top. During transport the radar was lowered to just above the Bimini top and secured with ratchet straps as in Figure 3. While underway, the carbon fiber pole was extended, and the radar's base plate was secured to the gunwale in two locations on either side of the boat. When extended, the radar was approximately 3 meters above the water. This allowed for better "visibility" of the radar, as well as provided safety for the boat's passengers walking on the deck. The RadarNav system ran on a Linux laptop stowed under one of the aft passenger seats. A separate computer running a Windows operating system was used to control the radar. This was stowed under the other aft passenger seat, and was completely separate from the RadarNav system with the exception that the RadarNav system would connect to the controller computer via SSH in order to retrieve the digital radar data. The radar data was also collected during the real-time demonstrations. This data was used for post-test analysis and for testing proposed improvements to the system. A single axis gyroscope was used to aid in radar stitching, and a GPS receiver was used to collect position data. The GPS was not used by the RadarNav system, only for post-test analysis.

Figure 3. RadarNav boat – 22’ Boston Whaler with radar mast.

2.2 DATA COLLECTION HARDWARE

For the radar data collections, a capture hardware system was required. This system needed to be easy to use, and capable of connecting to multiple radar systems with different infrastructures and interconnects. Some systems transmit data through a CAN bus, others through standard Ethernet, and some had a video distribution network for the radar display. The video display was the most common data transmission element, and typically it was a standard VGA or DVI interface. Therefore, it was decided that the easiest way to capture the radar data across various systems was to record the video going to the display. The radar capture hardware consisted of an AV.io HD video grabber from Epiphan Video connected to a Raspberry Pi 3B+, with a USB flash drive for storage. Operationally, the cable that provides video from the radar control unit to the display was disconnected, and plugged into the input of a DVI splitter. The DVI splitter allowed one output to remain connected to the display for uninterrupted vessel operations, and another output to feed the radar capture hardware. Power was supplied through two AC-DC power adapters (one for the Raspberry Pi, and one for the video grabber). Figure 4 shows an illustration of this configuration.

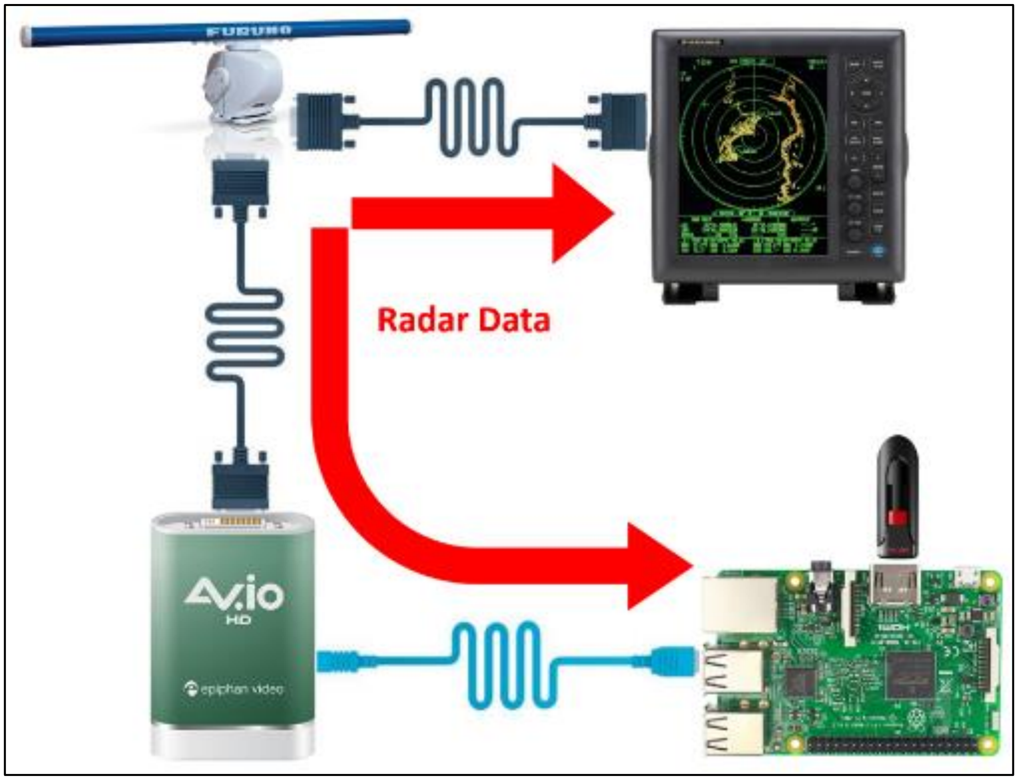

Figure 4. Block diagram of radar data collection hardware.

The video was captured as individual images at a rate of five frames per second. The recording software was configured to run as soon as the Raspberry Pi booted up. Capturing individual images made the processing easier on the back end, since these images were compared to the simulated radar images from the RadarNav system. Additionally, if the system failed at any point, only a few images might be lost, rather than the entire video. Once the video cables were properly connected, the user only had to plug in the power cables to start capturing. When finished, the user could just unplug the system. The captured images did require some processing before they could be used in the RadarNav system. Figure 5 shows a typical radar display screenshot that was captured, and the cropped version with the menus and extraneous information removed.

Figure 5. Radar display screenshot cropped to radar plot by itself.

Since each display was a little different, each set of images required slightly different processing before being used with the RadarNav algorithm. Fortunately, all the images in each set required the same cropping, so this was able to be done with a script.

The biggest problem with this method of data collection is the possible “noise” in the images from other things being displayed on the screen. This could be a cursor, or an overlay on the radar display; or it could be another window moved in front of the radar display temporarily. In one case the computer restarted and all of the splash screens and startup screens needed to be removed. Unusable images were removed, and the best effort was made to remove other images with a lot of noise. Figure 6 shows some examples of radar images with these “noise” artifacts.

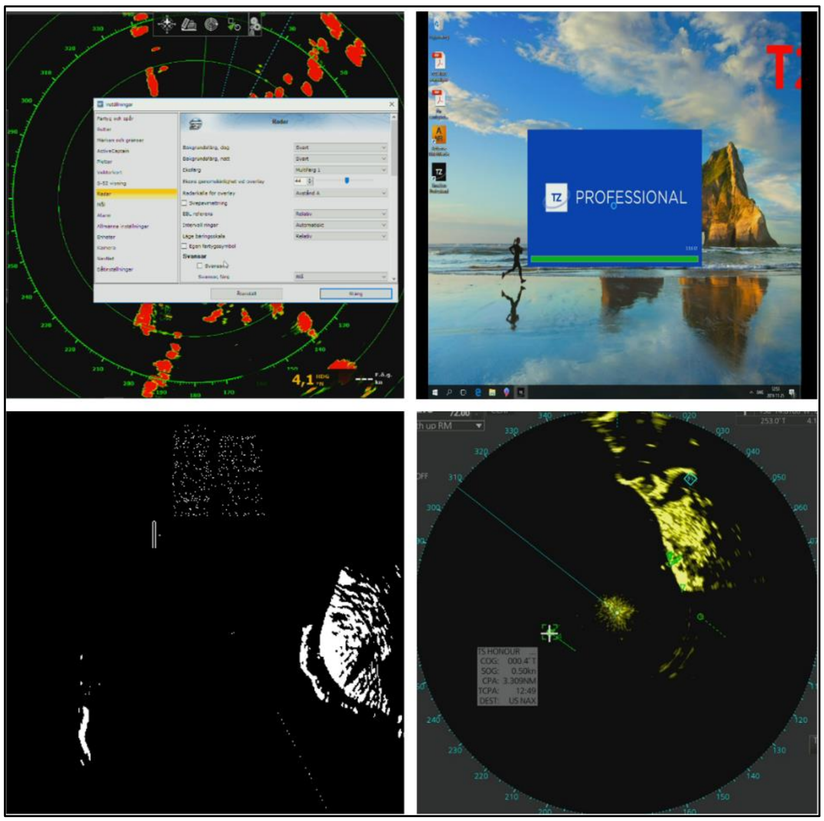

Figure 6. Examples of “noisy” radar images.

RadarNav depends on the ability to match these images to synthesized radar images. Since a cursor would not be present in the synthesized images, matching would be more difficult. While RadarNav is able to overcome a single cursor, each additional noise artifact will increase the actual and estimated errors. Unfortunately, removing the image creates a different problem. If there is no radar image, RadarNav will have a gap in its input data. This will increase the errors for each gap. RadarNav will recover once new radar images are available, but the error during those gaps may be higher. This is often presented as spikes in the error plots, although these gaps are not the only reason for these spikes.

2.3 POST-PROCESSING HARDWARE

Since there was no restriction to operate in real time, the captured radar data could be run through the RadarNav algorithm on almost any computer. However, looking towards the future, the data was run on a laptop similar to the one used for the real-time demos. Several types of mini-computers were also used just to see if they were capable of operating RadarNav in real time. A more detailed discussion of these results can be found in section 6.1 of this report.

This page is intentionally blank

3 SOFTWARE ARCHITECTURE

The RadarNav software runs on top of the Robot Operating System (ROS), on a Linux computer. Although it has "operating system" in its name, ROS is actually middleware – a collection of applications and frameworks useful for robotic control and communication. In the case of RadarNav, ROS was chosen as an easy means of communication between RadarNav's various modules since it was already well understood by project engineers. A simplified version of RadarNav's block diagram is provided in Figure 7.

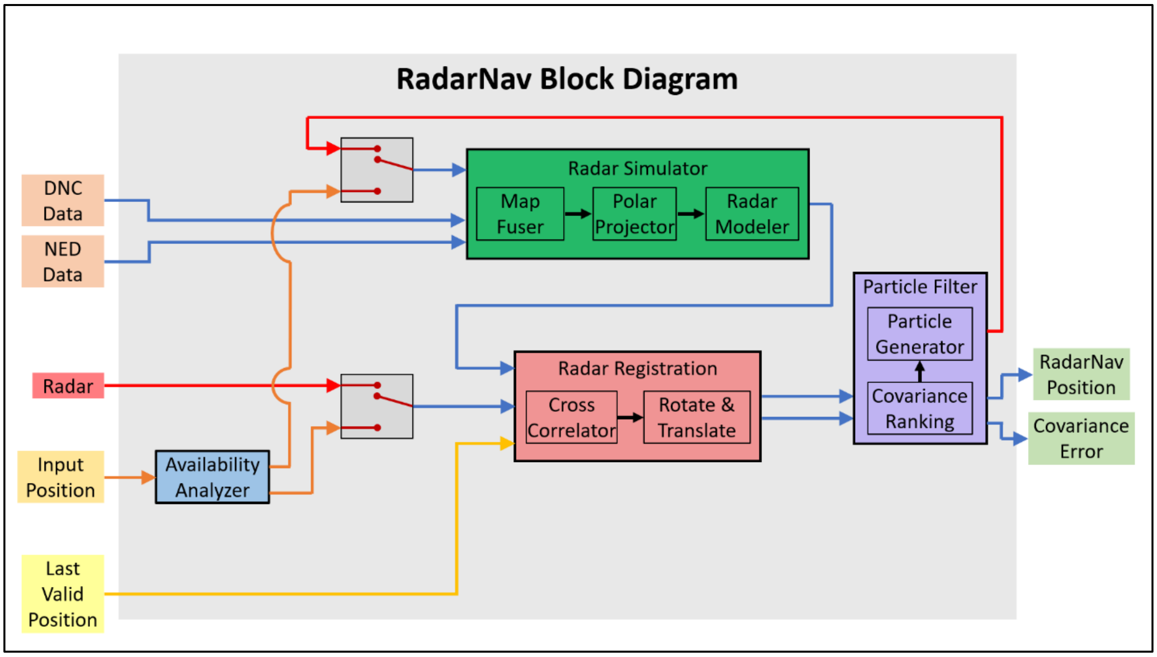

Figure 7. RadarNav software block diagram.

The main components of RadarNav's software are the Radar Simulator, Radar Registration, Particle Filter, and Availability Analyzer. Each of these blocks will be discussed in more detail in the following sections. The RadarNav algorithm is an iterative process that cycles through each time a new radar image is received from the radar system. Starting with the particle filter; particles are created for evenly distributed positions across the area of interest. The radar simulator synthesizes a radar image for each position (or particle) created by the particle filter. The radar registration module compares these synthesized radar images to the real radar image. Each of these comparisons represent a latitude and longitude position in the real world, and a particle within the RadarNav system. The radar registration module assigns a covariance to each particle based on how well the images match. Next, the particle filter ranks the particles according to their covariances and removes any particles whose covariance falls below a specified threshold. The particle filter then generates new particles and sends their positions to the radar simulator to synthesize new radar images to compare to the next real radar image that is received. Eventually the particles converge and the most probable position estimate is chosen as RadarNav's position solution.

3.1 RADAR SIMULATOR

The radar simulator is responsible for converting the map data into a synthesized radar image centered at a specified point. This module also allows for limited modeling of the radar being used, and other corrections that aid in matching the synthesized images to the real radar images.

The first step in the radar simulation process is to fuse data from multiple maps. RadarNav uses Digital Elevation Models (DEMs) from the 3D Elevation Program (3DEP), provided by the United States Geological Survey (USGS), and Digital Nautical Charts (DNCs) from National Geospatial-Intelligence Agency (NGA). More information on the map sources can be found in section 0 of this report. The elevation data provides information about the terrain and topography of the coastline/landmass. The DNCs provide details about man-made structures and fill in gaps in the elevation data. These gaps may be due to the fact that the elevation is below sea level so the area may be considered water, or simply just missing information in the elevation data. Figure 8 shows what the North end of the San Diego Bay looks like in the DEM by itself. Figure 9 shows the same image after the DEM was fused with the DNC. The black area in both figures is considered water, which is the only area that RadarNav considers a possible location for the vessel. The top of Figure 8 shows what appears to be water, however, there is no water in that location in the real world. In Figure 9 this area is filled in, and will no longer be considered water by RadarNav. Additionally, the fused image in Figure 9 shows multiple jetties and piers that are not present in Figure 8. These structures greatly improve RadarNav's position estimate because man-made structures typically have straight edges and square corners. This means they will stand out against the natural terrain, so they provide more unique features to match between the images. This is especially true of taller structures that the ship's radar will have an easier time "seeing".

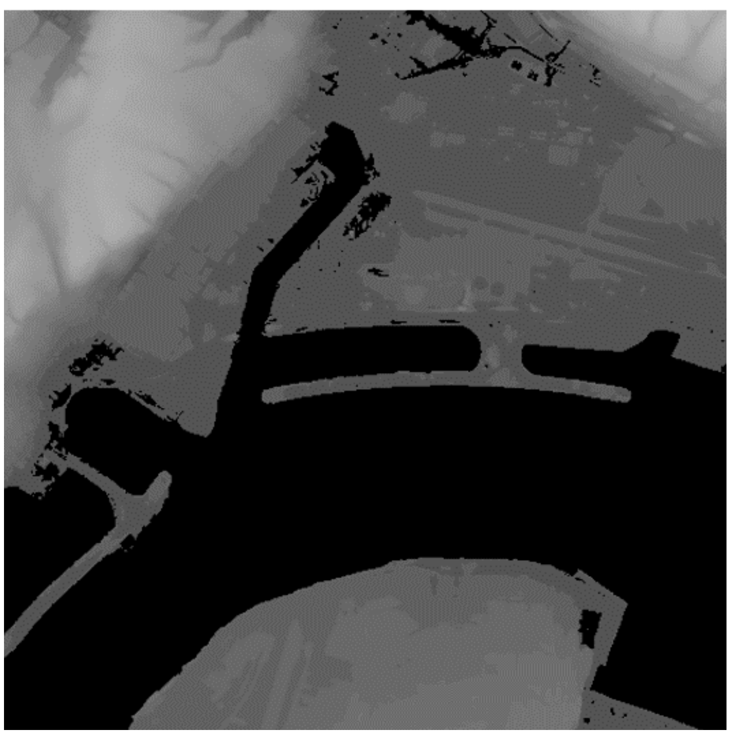

Figure 8. Typical image from USGS DEM. North end of San Diego Bay.

Figure 9. Typical DEM fused with Digital Nautical Chart. North end of San Diego Bay.

The second step in the radar simulation process is the polar projection. Standard 2D maps like those shown in Figure 8 and Figure 9 do not look like images generated by a radar system. The reason for this is that radar systems generate their images by preforming a 360° sweep of transmitted and received radio waves. On ships, these radio waves travel more-or-less parallel to the surface of the earth. The images that are produced are based on the time of flight of these radio waves, so the perspective of the images is from the ship, looking out rather than the bird's eye perspective of standard 2D maps. Obviously matching radar images directly to 2D map images wouldn't work. This is where the polar projector section of the radar simulator comes in. Figure 10 shows the steps that the polar projector takes to convert the 2D source map images into synthesized radar images. First, the fused 2D map image is converted to polar coordinates with the assumption that the radar is located at the origin. Next the horizon and occlusion effects are estimated in the polar map image. Then the polar map image is converted back to rectangular coordinates. At this point, the map image looks like an image that would be produced by a radar system.

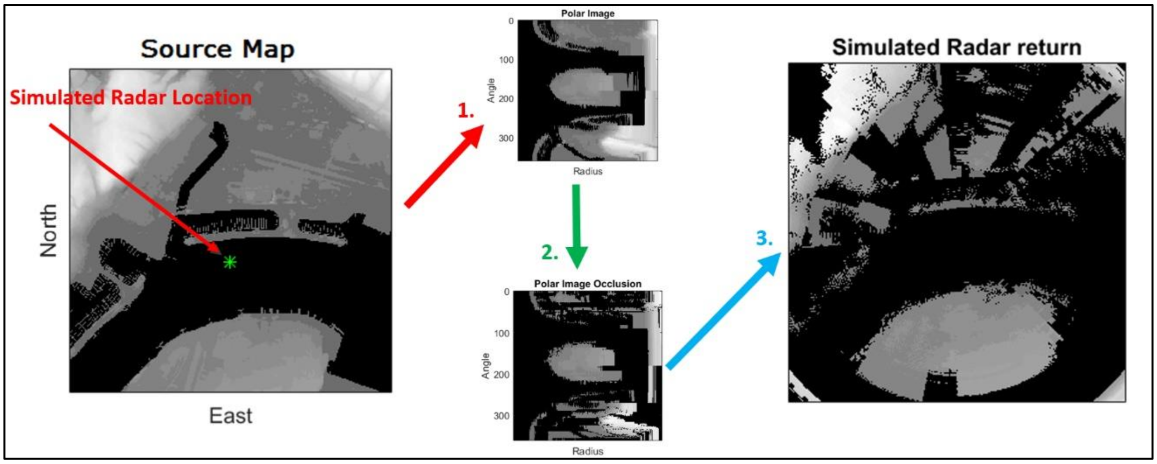

Figure 10. Work flow of polar projector module.

Once a radar image has been synthesized, more image processing can be done to add noise or other artifacts that help model the radar system being used. Although this would be done after synthesizing the radar image, most of the radar modeling is actually done throughout the polar projection process. Radar effects that RadarNav models include the height of the radar above the water line, the resolution of the radar system, and the power that the radar system is using to transmit the radio waves. These last two parameters are often configurable on modern radar systems, so RadarNav needs this information to provide accurate position estimates. The height of the radar is especially important for modeling occlusions and estimating the horizon (especially on more powerful systems). Occlusions are features that the radar wouldn't be able to "see" because the radio waves are being blocked or reflected before reaching those features. The horizon can also cause occlusions because the radio waves don't bend around the curvature of the earth. Since radar systems must have line of sight to any potential targets, the horizon represents the functional limit of radar in the open ocean (assuming the radar has sufficient power). Beyond the horizon, there is nothing left for the radio waves to reflect off of. This is illustrated in Figure 11. Figure 11 also shows that the distance to the horizon is related to the observation height. This is why the radar's mounting height is a critical piece of information for RadarNav.

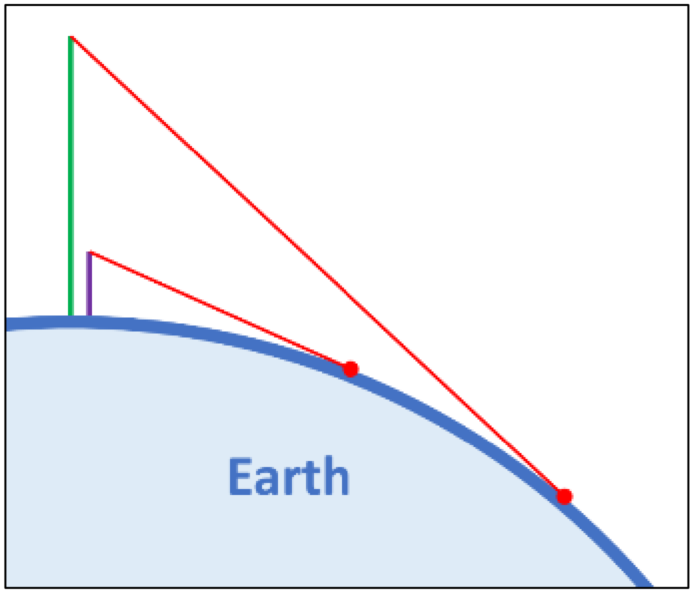

Figure 11. Illustration of distance to horizon.

In addition to occluding features beyond the horizon, the horizon can alter the apparent height of features. As Figure 12 illustrates, tall objects beyond the horizon are still within the radar's line of sight, but their height will appear to be shorter due to the curvature of the Earth. This would make matching synthesized and real radar images difficult. As part of the radar modeling module, RadarNav corrects for these height differences by calculating the distance to the horizon based on the radar's mounting height, and adjusts the elevations of features beyond the calculated horizon by subtracting from those features' elevations on the elevation map.

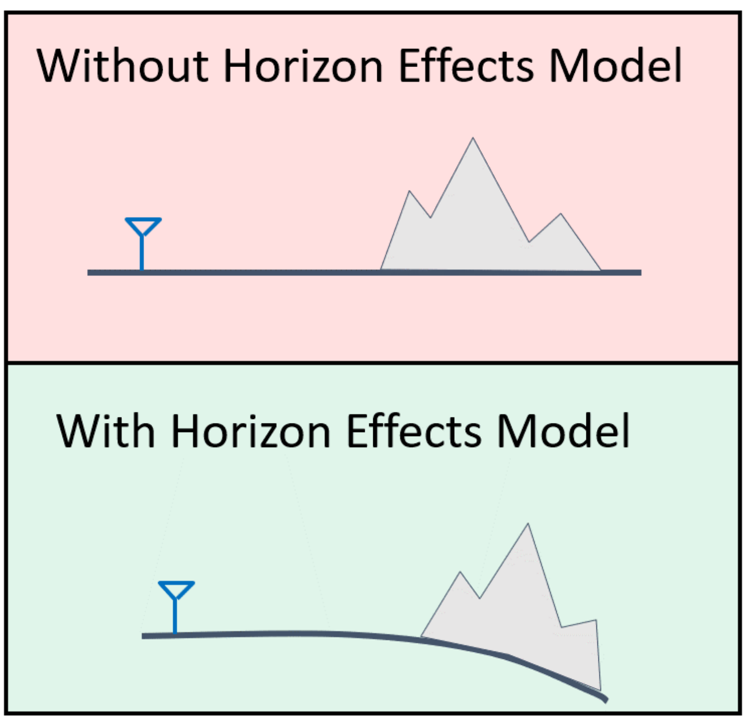

Figure 12. Illustration of horizon effects model.

3.2 RADAR REGISTRATION

The radar registration module does the bulk of the processing in the RadarNav algorithm. It is responsible for estimating translative effects between two similar images, registering an image based on those effects and calculating the likelihood that two images match. Registering an image is the process of adjusting it for rotational, translational and scaling differences based on a reference image. For RadarNav, this process compares an incoming radar image (the “sense” image) to a synthesized radar image (the “reference” image). Registering the sense image ensures that the incoming radar image and reference image have the same scale and are similarly oriented in terms of cardinal directions. This improves the phase correlator’s ability to match synthesized radar images to radar images from the real world. Registering the image also provides information about the change in the radar’s orientation in the real world, from one radar sweep to the next. The registration effect is shown in Figure 13. The strip of land in the real image resembles a forward slash character (/) while the land in the synthesized image looks more like a backslash character (\). This might happen when the map is orientated differently than the radar in terms of cardinal directions. After the registration process, the real and synthesized images are a much closer match.

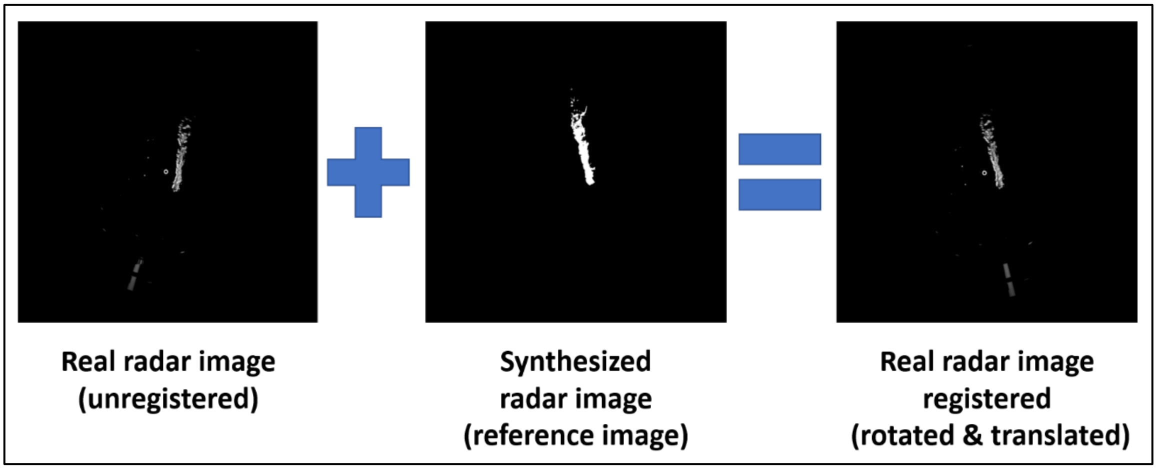

Figure 13. Radar registration between real and synthesized radar images.

The radar registration module uses a subset of cross correlation called phase correlation to estimate the translative effects that must be applied to the image. Phase correlation uses a frequency domain representation of the image and isolates the phase information from the Fourier-space data. The magnitudes of all frequencies are set to one prior to the correlation step. In its current configuration (and for all the all the results presented in this report), RadarNav limits the images to 256 x 256 pixels which results in a 65536-dimensional vector. The phase correlation is calculated using the log of a Gaussian probability as shown in Equation 1.

Equation 1

$$\log P(\mathbf{y}) = \mathrm{N}\log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) - \sum_{N}\frac{(y_n-\mu_n)^2}{2\sigma^2}$$

The major benefit of phase correlation over standard cross correlation is that it is more resistant to errors that may be caused by occlusions. A block diagram of the radar registration algorithm used by RadarNav is shown in Figure 14. This algorithm was developed by (Jignesh N Sarvaiya, 2009).

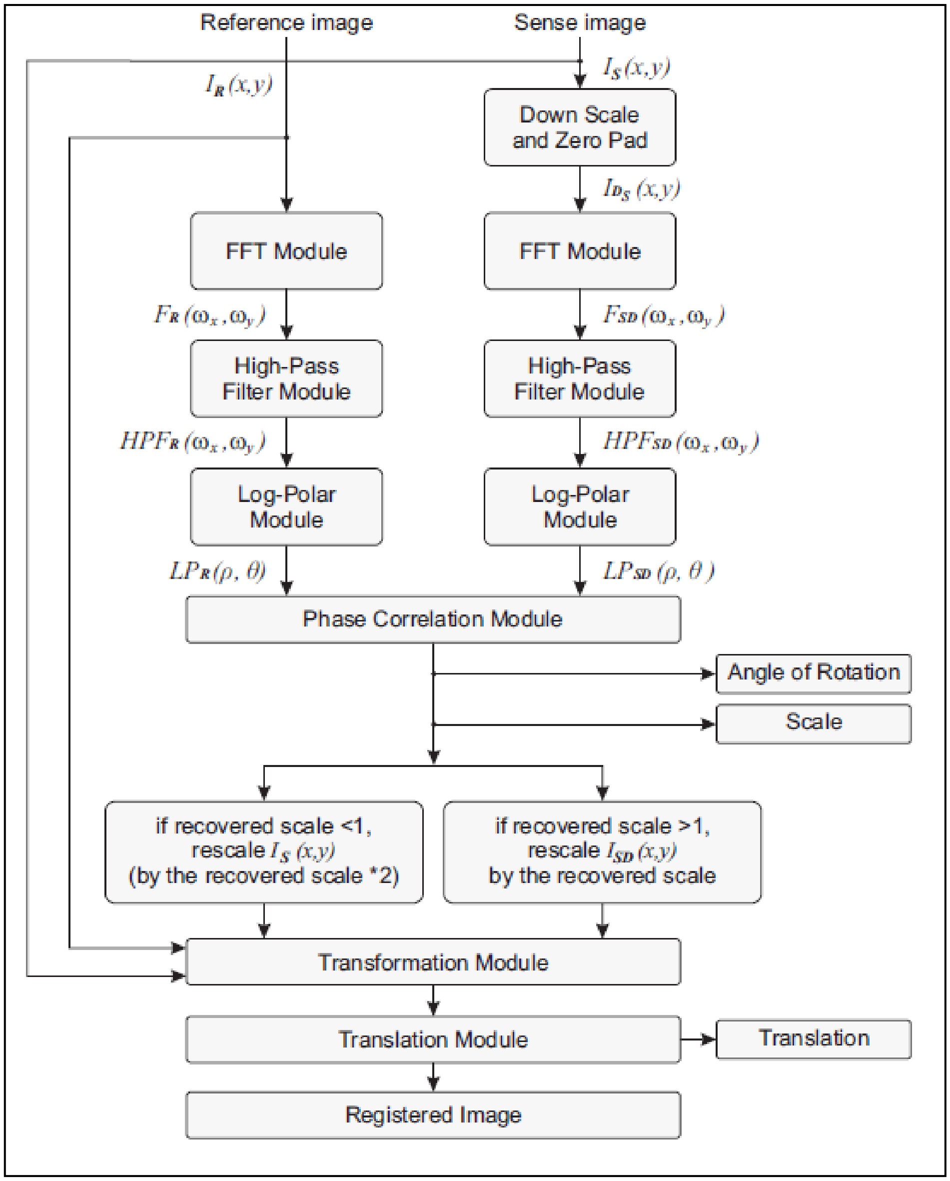

Figure 14. Block diagram of RadarNav's radar registration algorithm from (Jignesh N Sarvaiya, 2009).

The radar registration module also provides a covariance estimate that describes how well the real radar image matched to the synthesized radar image. This covariance serves two purposes. First, it allows the particle filter to organize and remove particles based on how well they match the real radar image. Secondly, the covariance of the particle that is chosen as RadarNav's position solution serves as a confidence metric for RadarNav's users. This gives the user some idea of how confident RadarNav is with its solutions, and how much the user should trust it.

3.3 PARTICLE FILTER

RadarNav's particle filter is responsible for ranking the particles (or possible positions) from the radar registration module, removing particles with a covariance below the specified threshold and generating new particles (or positions) to send to the radar simulator. Figure 15 shows the progression of the particle filter as it begins to converge on the position solution. The black outline is the coast of San Diego and the purple markers represents the positions of the particles. Notice that only particles in the water are considered valid. As new real radar images are input into the particle filter, the particles begin to get lumped together into clusters which are then thinned out and removed until only the most reasonable position estimate remains.

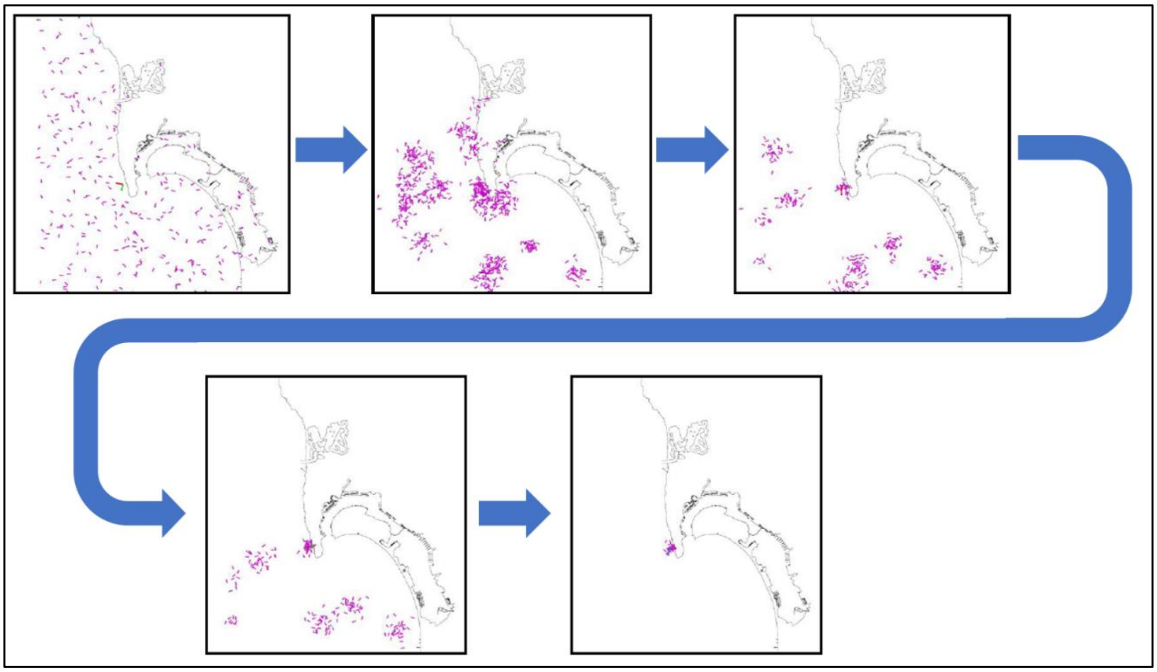

Figure 15. Illustration of particle filter convergence.

RadarNav uses a limited number of particles, with the maximum number defined in a configuration file. For the results discussed in this report, a particle is generated for every 13.5 square kilometers in the initial wide search. When clusters are defined, the maximum number of particles was set to 32 for each cluster. RadarNav has the option of specifying an initial position. In an operational setting, the last known valid position would be used. This reduces the time it takes RadarNav to converge on a reasonable position solution. For the purposes of the work presented in this report, an initial position was never used – this is known as a cold start. For a true cold start any position in a body of water, anywhere in the world would be a possible position solution. However, this is not practical, so during this effort, the search area was limited to a couple map tiles (in most cases only a single map tile was used). The first image of Figure 15 shows that the particles are evenly distributed throughout the map area on a cold start. Each time a new radar image is received, the particle filter removes particles with covariances below a specified threshold, and creates new particles clustered around the remaining particles that had a high enough covariance. With each new radar image, clusters around the more probable positions become denser, and less likely locations fade away. Eventually the particles converge on the most likely position. This becomes RadarNav's output solution; and the covariance related to the synthesized radar image at that position becomes the real-time estimated error that is provided as a confidence metric for that position.

During a cold start the time-to-first-fix depends on the processor, number of valid particles, radar's update rate (sweep rate) and the number of possible positions on the provided map (anywhere covered by water within the map boundaries). The example in Figure 15 is from the San Diego dataset. In that scenario RadarNav took about 60 seconds to converge on a reasonable position. This was done using a Panasonic CF-54 Toughbook laptop with an Intel i7-6600U, 2.60GHz processor and 16GB RAM. Figure 16 shows a map that illustrates this cold start and convergence to a position fairly close to the actual starting position. In this case, the initial position estimate was about 4.7Km away from the actual starting position, but RadarNav was able to get to a more reasonable solution rather quickly.

Figure 16. Map of San Diego, CA showing RadarNav cold start.

Figure 17 shows this same cold start convergence information in the form of a plot of position error (red line) versus time. This plot also shows the estimated error (green area). Notice that the estimated error over-bounds the actual error. This is done on purpose because the estimated error is meant to be used as a measure for how much confidence RadarNav has in its solution. At the beginning of the cold start, a user would see an estimated error of over 6Km and know that the position solution should not be trusted. However, once RadarNav converges on a reasonable solution, the estimated error is greatly reduced.

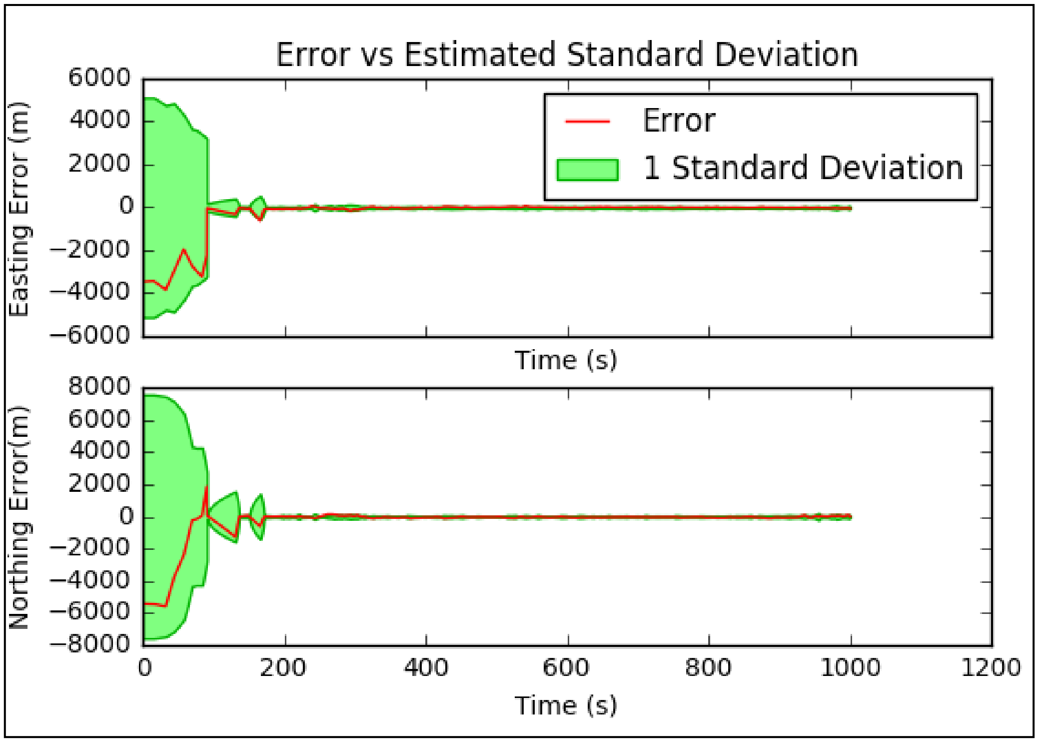

Figure 17. Error vs. covariance estimation during convergence.

Figure 18 shows the same data plotted in Figure 17, but cropped and zoomed in to show the data after convergence. Ideally, the error should never be larger than the estimated covariance in order to keep the confidence metric reliable.

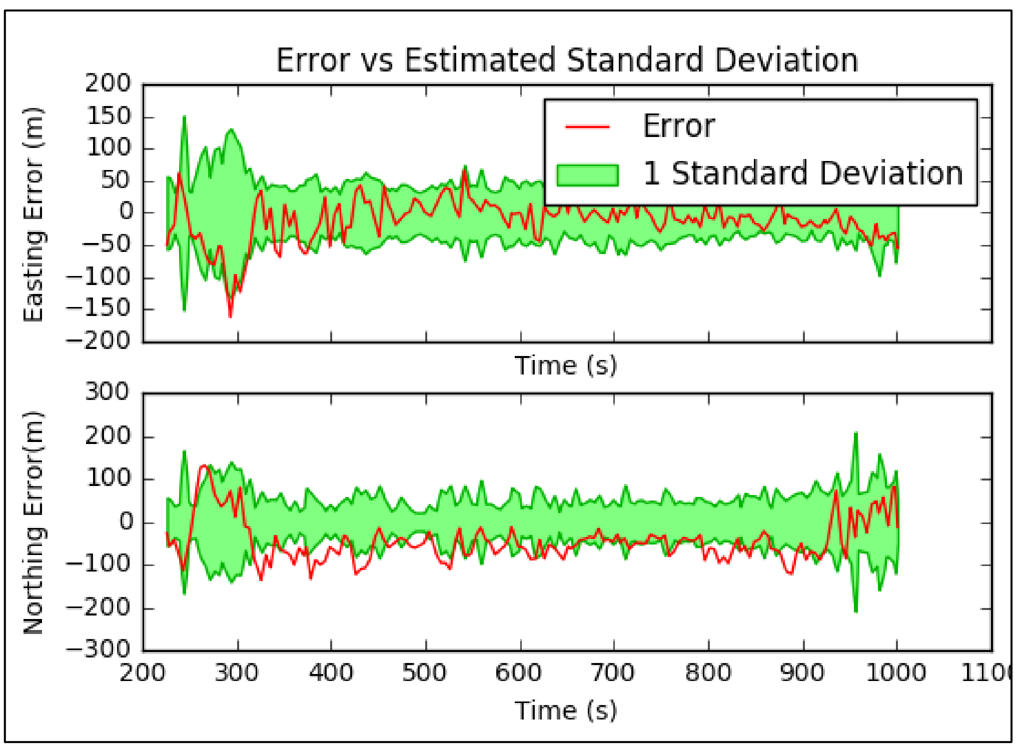

Figure 18. Error vs. covariance estimation after convergence.

3.4 AVAILABILITY ANALYZER

RadarNav's availability analyzer is a mission planning tool that gives users the ability to determine how well RadarNav is expected to work given certain radar characteristics and the geographic features in the area of interest. Users can load a single point or an entire track (multiple points) into the availability analyzer and it will provide an estimate of RadarNav's position error at each point. This is referred to as the predicted error. The availability analyzer is not technically a part of the RadarNav system, but an additional tool that utilizes most of RadarNav's modules. The availability analyzer works by using a synthesized radar image in place of the "sense" image from a real radar system. This is the instance where the radar registration module compares two synthesized images. Once a radar image is synthesized for each of the latitude and longitude positions of interest, radar specific effects are modeled and then gradients are calculated with respect to each position. Next the Cramér–Rao Lower Bound (CRLB) is calculated for each position. The CRLB is the minimum estimate for the variance of an unbiased estimator. In the case of RadarNav, the estimator is the phase correlator, and the CRLB represents the minimum error achievable by RadarNav for the given position. The CRLB is then scaled with noise to match real-world performance and provide a realistic idea of RadarNav's accuracy for the positions of interest. The predicted error provided by the availability analyzer represents the best possible accuracy RadarNav can achieve under the provided circumstances. Real-world errors will be larger and the real-time estimated errors will be even larger. This is done on purpose because the availability analyzer is meant as a planning tool. If the best possible performance still isn't good enough, the route should be changed, or an alternative navigation system should be used.

Figure 19 shows a heat map plot of the availability analyzer's output for the entire San Diego area. Rather than a single position or a track with multiple positions, Figure 19 plots the expected performance of RadarNav for every point on the San Diego coast. This was done with the assumption that the Furuno FAR-2117 and the 4' XN12AF antenna were used. Inside the bay where there are many unique features to map to, the expected RMS predicted error is around 10 meters (dark orange areas). This error increases as the coast gets further away until it reaches about 2,000 meters (light purple areas). At this point the expected predicted error increases exponentially and is so large that RadarNav would be considered useless (grey and white areas). According to the availability analyzer RadarNav should have less than a 200m RMS error up to about 6KM from shore.

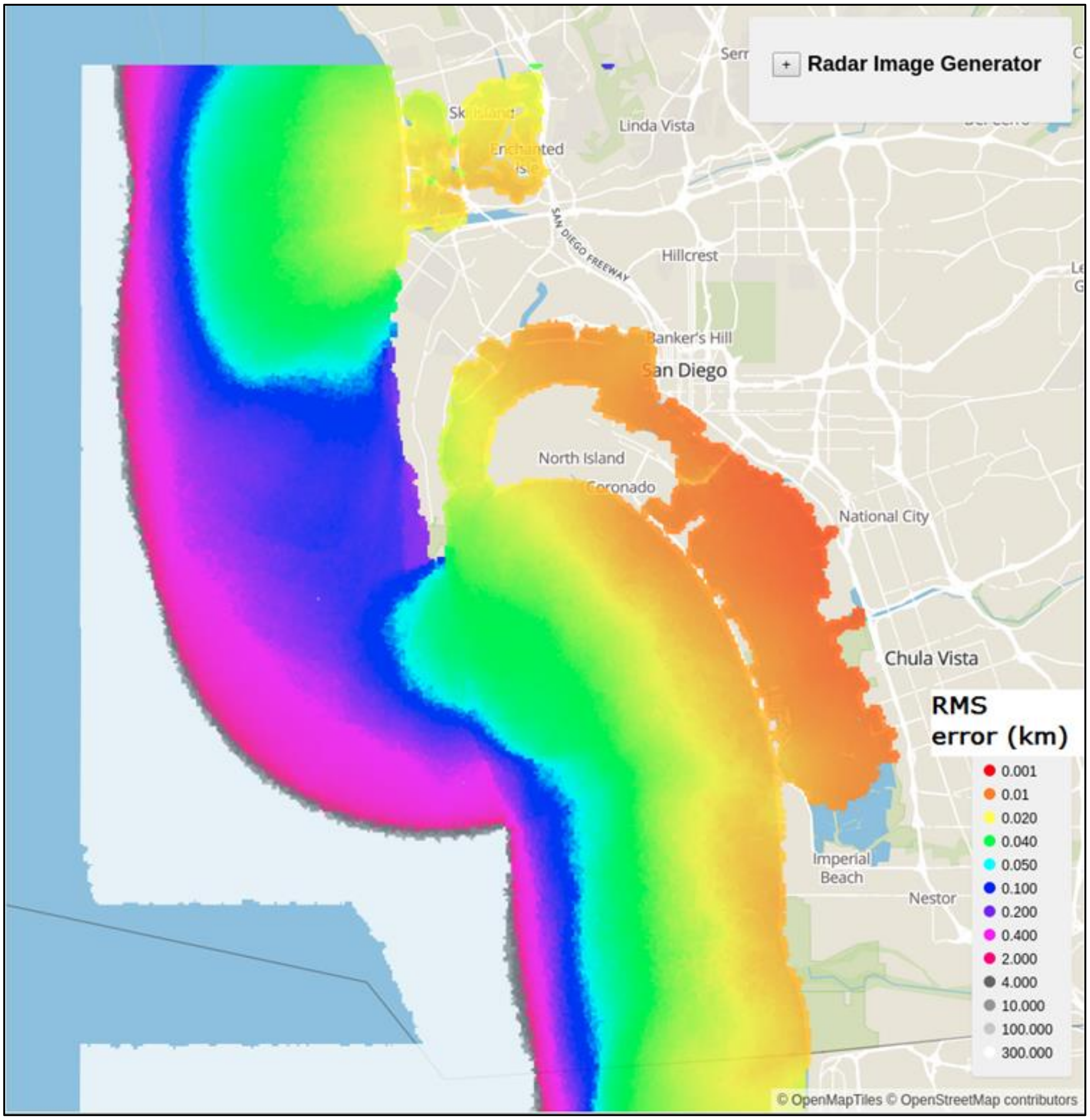

Figure 19. Availability Analyzer's outputs for San Diego (Street map Mapbox©, OpenStreetMap©).

The center of Figure 19 shows an area where RadarNav's performance suffers near the shore. This is marked by the blue and dark purple areas right next to the shore in the center of the plot. In the real world, this area is marked by cliffs and hillsides without many man-made or unique geographic features for RadarNav to match to. Figure 20 shows an image of this area from Google Earth. Figure 19 actually shows that the predicted error is worse close to shore (purple area), then improves slightly (blue area) as the shore gets farther away. This is because more unique features can be "seen" by the radar as it gets further away from the shore.

Figure 20. Google Earth image of Sunset Cliffs area of San Diego coastline.

This page is intentionally blank.

4 MAPS

RadarNav can be configured to work with any digital map that has coastline information with realistic dimensions/scale. However, the quality of the maps has a direct impact on the positional accuracy that RadarNav is able to achieve. For the purposes of the data presented in this report, only publicly available map sources were used. In an operational environment, RadarNav could utilize better maps from possibly classified sources.

4.1 MAP SOURCES WITHIN THE UNITED STATES

The main source of maps used for data sets within the United States and Hawaii were Digital Elevation Models (DEMs) from the 3D Elevation Program (3DEP), provided by the United States Geological Survey (USGS). These DEMs were referred to as the National Elevation Dataset (NED) at the beginning of this effort, and may still be referenced as such throughout this report. The 3DEP provides the elevation layer of The National Map from the USGS, and consists of several DEMs of varying resolutions. The DEMs with 1/3 and 1 arcsecond resolutions were used for RadarNav because they are high quality and have nationally seamless coverage. The 3DEP also contains newer DEMs with 1/9 and 1/30 arcsecond resolutions, however, they do not have complete coverage, and could not be used in all areas of interest for this report. The National Map and the 3DEP layers can be accessed at the link in ((USGS), The National Map Viewer, n.d.).

RadarNav also uses Digital Nautical Charts (DNCs) to enhance the elevation maps. These DNCs are produced by the National Geospatial-Intelligence Agency (NGA). They are vector-based digital products that contain features significant to maritime navigation. For the purposes of this research effort, only the unclassified, publicly available DNCs were used. At the time this report was written, these DNCs could be downloaded from the link in ((NGA), n.d.).

4.2 MAP SOURCES OUTSIDE THE UNITED STATES

The National Map from the USGS provided map information for data sets in the United States. However, these maps do not cover areas outside of the United States. For the Sweden data sets, the elevation data was provided by the USGS through their Earth Explorer website ((USGS), EarthExplorer, n.d.). The website provides public access to the Shuttle Radar Topography Mission (SRTM) dataset. The SRTM dataset was collected by NASA on a 2000 space shuttle mission that provides global digital elevation maps at 1 arc-second resolution.

NGA's DNCs were also not available for the Sweden data set, so the coastline data was provided by the National Centers for Environmental Information (NCEI); which is an organization under the National Oceanic and Atmospheric Administration (NOAA). NCEI maintains the Global Self-consistent, Hierarchical, High-resolution Geography (GSHHG) database ((NCEI), n.d.) which is a global coastline dataset that can be used by the public.

4.3 SCALE AND RESOLUTION

The quality of a map depends on the spatial accuracy of objects on the map and how closely those objects represent the real world. A map's scale specifies the amount of reduction between the real world and the graphical representation (i.e. map). Scales are commonly expressed as ratios (e.g. 1:20,000). This reduction is achieved by removing information, so the accuracy of the map decreases as the scale increases. In general terms, resolution of data is the smallest perceivable change between adjacent points. With paper maps, the resolution is tied to the scale because the paper is always the same physical size. As the scale increased (more reduction) the resolution decreased because information was removed, so there are fewer pieces of information left to represent the same features.

With raster representations of maps, these pieces of information are called pixels. Pixels form a grid over the map area with each pixel representing the smallest piece of information available. The resolution of raster data is the amount of information contained in each pixel. Since pixels are square (usually) and maps are concerned with distances between features, the resolution is represented as some unit of distance per pixel (e.g. meters per pixel or m/p).

The Digital Elevation Models (DEMs) are raster representations of the Earth's surface. Each DEM consists of tiles that correspond to topographic quadrangles of different scales which align to the Universal Transverse Mercator (UTM). Since the Earth is round, the USGS likes to specify the distance component of their map's spatial resolution in terms of degrees/minutes/seconds of an arc instead of meters. An arcsecond is 1/3600 of a degree or 1/1,296,000 of a complete rotation (360°). Since the Earth's circumference is $40.075e^6$ m, one arcsecond is about 30.9 m (this is only an approximation at sea level along the equator). Table 1 provides approximate conversions between arcsecond and meter resolutions for the DEM layers used in The National Map from USGS. The 1 arcsecond DEM is accurate down to about 30 meters, so any features smaller than 30 meters would not be available in this DEM. Figure 21 provides an idea of the coverage of each of the DEM layers at the time of this report.

Table 1. Approximate relationships between arcsecond and meter resolutions.

Arcsecond Resolution of DEM Layers	Approximate Meter Resolution of DEM Layers
2 arcseconds	60 meters
1 arcsecond	30 meters
1/3 arcseconds	10 meters
1/6 arcseconds	5 meters
1/9 arcseconds	3 meters
1/30 arcseconds	1 meter

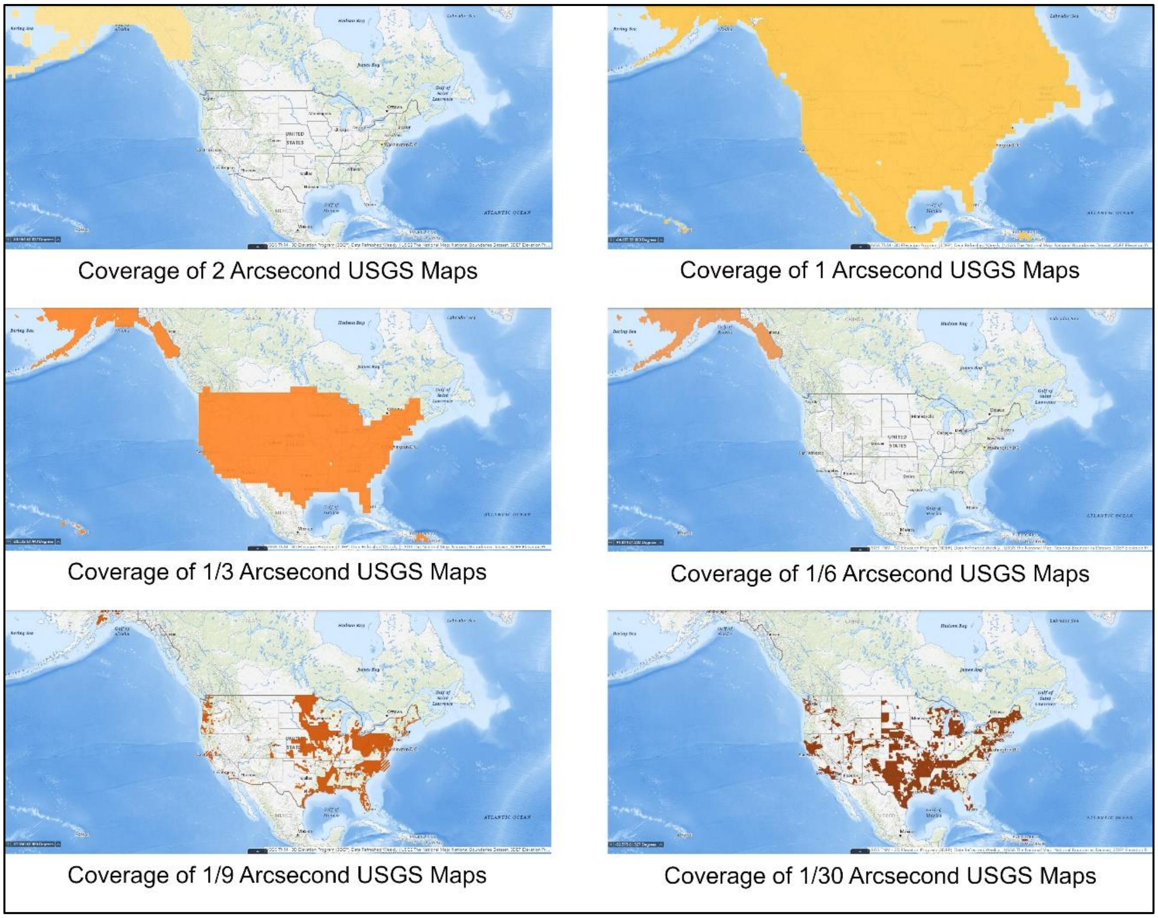

Figure 21. Coverage areas of the DEM layers of The National Map from USGS.

While the quality of the maps is important to RadarNav's accuracy, the quality of the radar data is equally important. Since RadarNav depends on matching features between images, the image with the lowest quality or highest meters per pixel (m/px) will be the limiting factor. The m/px resolution of the maps will carry through the map fusion and radar synthesis processes, but if the m/px resolution of the radar data being input to the system is lower, RadarNav will lower the quality of the synthesized images in order to properly match.

This page is intentionally blank.

5 TESTS AND DATA COLLECTS

The data presented in this report was collected from a variety of real radar systems on different vessels, in different locations. RadarNav was tested in both real-time and post-processed scenarios. However, the real-time scenarios were only to prove that RadarNav was capable of operating in real time. The radar images were collected during these live test events and the information presented here is the result of post-processing using the latest version of the RadarNav algorithms. Therefore, the variations in performance are due to geographic locations and individual radar systems; and were not impacted by changes to the software or algorithms. The data collected specifically for post-processing was done with the help of volunteer vessels and crews. Using the same boat and radar system to collect data in all of these different locations would have been impractical due to time and funding constraints. Furthermore, using data collected from different vessels, with different radar systems shows that RadarNav is flexible enough to work with a wide range of systems. The difference in radar hardware also highlights the effect that radar quality has on RadarNav's accuracy. For clarity, the various types of error discussed in this report are defined as:

1) "Error" is the actual error and difference between RadarNav's position solution and the true position (according to GPS). This is given in meters.
2) "Estimated Error" is the real-time estimated position error that RadarNav supplies to users as a confidence metric. This is RadarNav's estimate of the difference between its own position solution and the true (GPS) position based on the covariance of the particle/position selected. This is given in meters.
3) "Predicted Error" is the estimated position error from the availability analyzer. This is not based on real radar data, but is RadarNav's estimate of how accurate it thinks its position will be with a given radar system, in a given area. This is given in meters.
4) "Pixel Error" is the same as the actual, estimated and predicted position errors, but describes the difference in terms of pixels rather than meters. This is important when comparing data sets that use maps or radar images with different meter per pixel resolutions.

The following sections discuss the performance of RadarNav for data recorded at various locations. In order to focus on the data and performance during transit, the cold-starts have been removed from these plots.

5.1 RED BEACH, MCCS CAMP PENDLETON, CALIFORNIA

The radar data from Red Beach dataset was collected as part of a live demonstration using the 22' Boston Whaler as described in Section 2.1. The radar in use was a Furuno FAR-2117 X-Band radar with a maximum power of 12Kw, and a maximum range of 96 nautical miles. The antenna in use was 6.5ft long. The data was collected by connecting to the computer that controlled the radar (via SSH over a wired network), and pulling the digital radar data onto the laptop running the RadarNav software.

Figure 22 shows a map of the Red Beach area with the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). All of the land is to the West of the Radar. The vessel starts by heading North for about 6Km, then turns around and heads back South. The estimated error is smaller when the radar can "see" more man-made structures near the South end of the track. The estimated error grows as the vessel moves North, because this area is relatively flat and devoid of unique features. As the vessel moves back to the South end, the estimated error gets smaller again. It is difficult to see, but the RadarNav position estimate has a fairly consistent bias in this data set. It is offset about 70 -90 meters to the West of the actual GPS track. The reason for this is not currently known. However, a similar bias is present in the other data sets where land is only on a single side. The offset is not always the same distance or even to the same direction, but it is present.

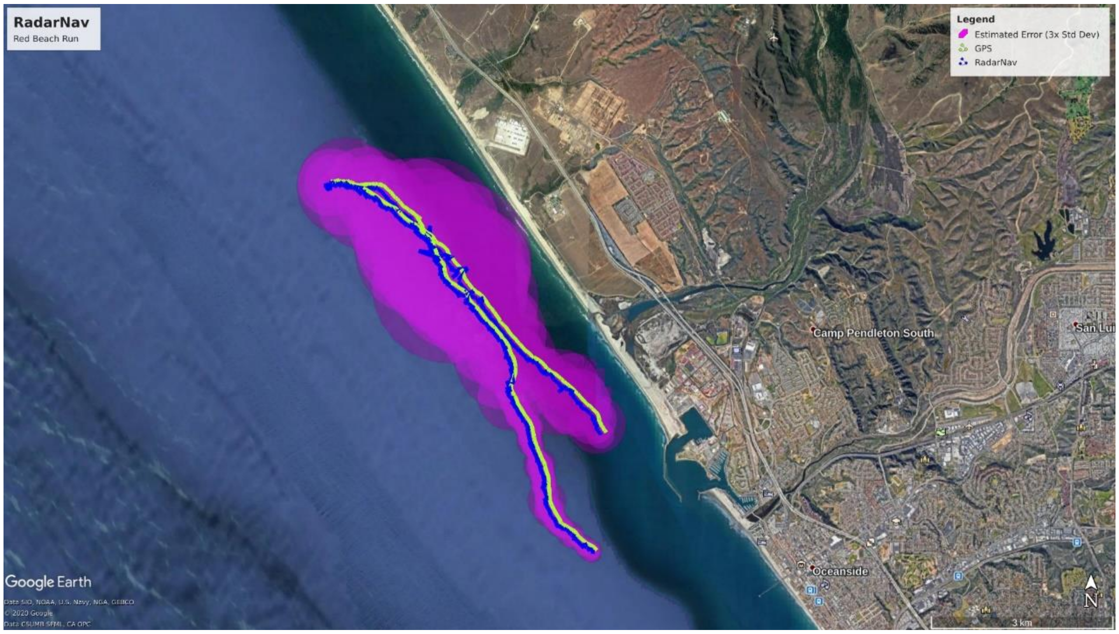

Figure 22. Map of RadarNav, GPS positions & estimated error for Red Beach.

Figure 23 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and the average, or Root Mean Square (RMS) of the actual error (solid blue line). This plot makes it easier to compare the different types of error associated with RadarNav. Figure 23 also shows that the estimated error has a relatively low baseline, but the many large spikes in the error estimate cause the error estimate (purple area) to look so large in Figure 22.

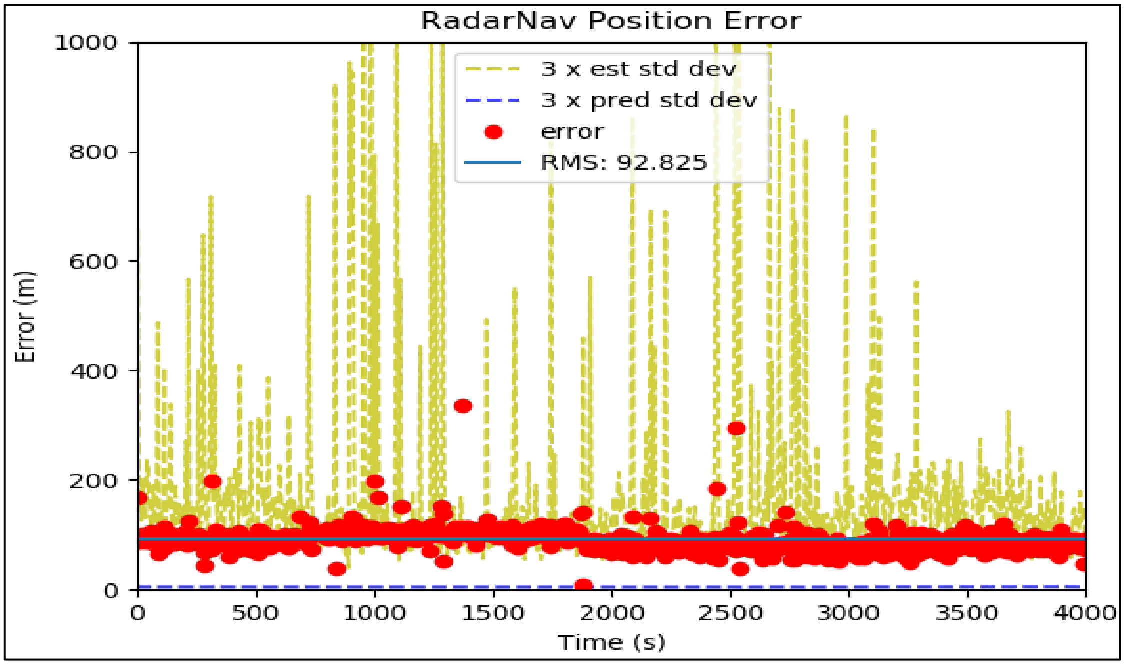

Figure 23. Plot of position errors vs. time for Red Beach data set.

Figure 24 shows the same data as Figure 23 but in terms of pixels instead of distance. Similar trends exist in both plots, but the scale of the Y-axis is the important piece of information here. The pixel error is useful when validating the performance of RadarNav because this is how RadarNav “views” the data. The conversion to error in terms of distance relies on the resolution of the radar system and the maps being used. The smallest unit of error RadarNav can have is a single pixel, but if each pixel is 100m across, then the distance error could never be less than 100m. The RMS pixel error for this data set is only 3 pixels (rounded from 2.4 because fractions of a pixel don’t exist), which is slightly better than RadarNav’s average pixel error.

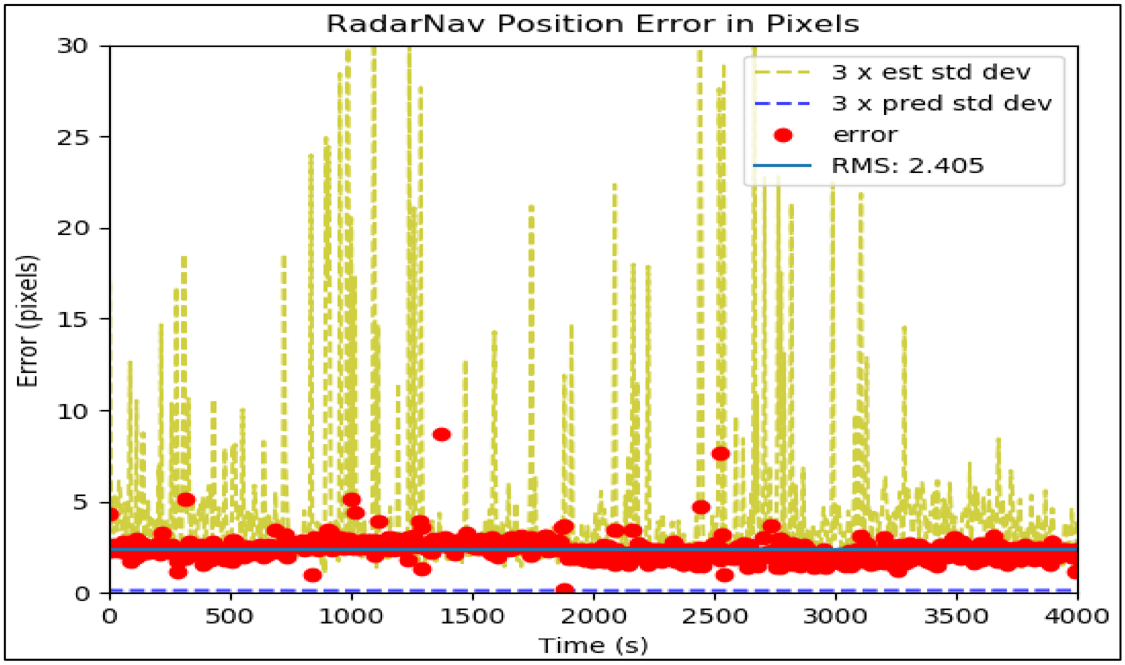

Figure 24. Plot of pixel errors vs. time for Red Beach data set.

5.2 SAN DIEGO, CALIFORNIA

The radar data from San Diego dataset was collected in the same manner, and using the same hardware as described in Section 5.1 for the Red Beach dataset. Figure 25 shows a map of the San Diego area with the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). For this data set, the vessel starts on the West side of Point Loma, and travels around the point, into the San Diego Bay. As the vessel moves into the bay, the estimated error is reduced because there is now land on both sides of the radar, as well as more man-made structures to help with image matching. It should be noted that in Figure 25, the estimated error ellipses (purple area) appear to be clipped off; especially at the beginning and end of the track. This was a graphics rendering issue with the ellipses in Google Earth, and did not have any impact on RadarNav.

Figure 25. Map of RadarNav, GPS positions & estimated error for San Diego, CA.

Figure 26 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and the RMS of the actual error (solid blue line). The predicted, estimated and actual error all drop after the radar moves around the tip of Point Loma and more land is “visible” since it is no longer being occluded by the relatively high elevation of Point Loma.

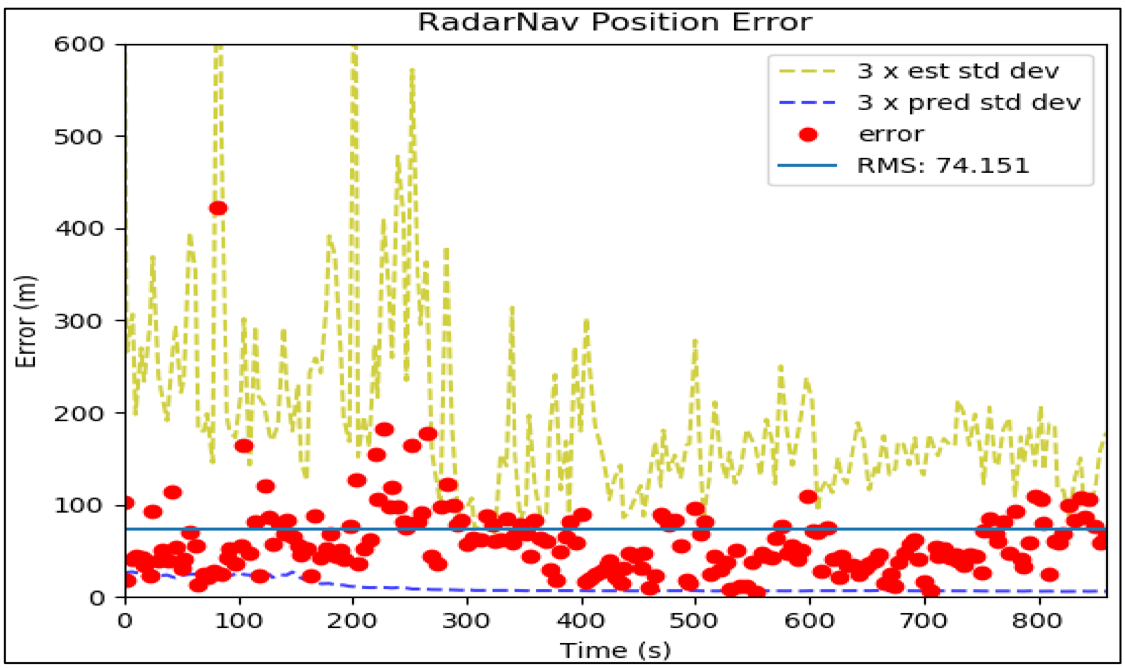

Figure 26. Plot of position errors vs. time for San Diego, CA data set.

Figure 27 shows the same data as Figure 26 but in terms of pixels instead of distance. The RMS pixel error for this data set is only 2 pixels (rounded from 1.9 because fractions of a pixel don’t exist). This is the lowest pixel error of all the data sets presented in this report.

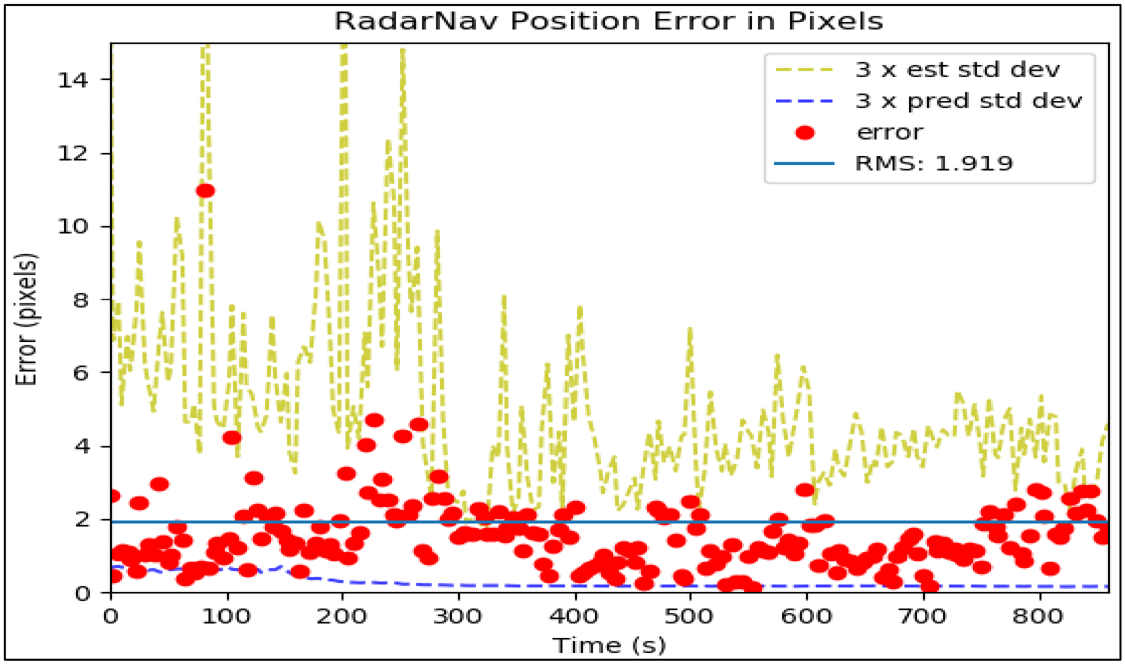

Figure 27. Plot of pixel errors vs. time for San Diego, CA data set.

5.3 NORFOLK

The radar data from the Norfolk dataset was collected for the purposes of post-processing. This data was collected from the NIWC Atlantic Position, Timing and Navigation (PNT) test boat by tapping into the network between the radar head unit and the console. The network data was captured using WireShark. A script was used to read the network packets, pull out the radar data and write it to a file with each row representing a full rotation of the radar. This file was then translated into image files which were processed by RadarNav's algorithm. The radar in use for this dataset was a Furuno DRS4A X-Band radar with a maximum power of 4Kw, and a maximum range of 48 nautical miles. The antenna in use was 3.5ft long.

Figure 28 shows a map of the South end of the Chesapeake Bay, near Norfolk, VA. with the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). The vessel starts at the South end of the track, and heads North along the Virginia coast. The error starts relatively small, but grows as the vessel moves away from land. The error begins to drop slightly as the vessel passes a small point, but then increases again toward the end.

Figure 28. Map of RadarNav, GPS positions & estimated error for Norfolk, VA.

The vessel continued North with a similar trajectory, beyond what is shown in Figure 28. However, the closest land is more than 4Km away at the northern-most end of the track in Figure 28. The radar was barely picking up any land at this point and soon the radar was not able to “see” any land at all. Figure 29 shows an example of a radar image from this data set where the radar was not able to “see” any land.

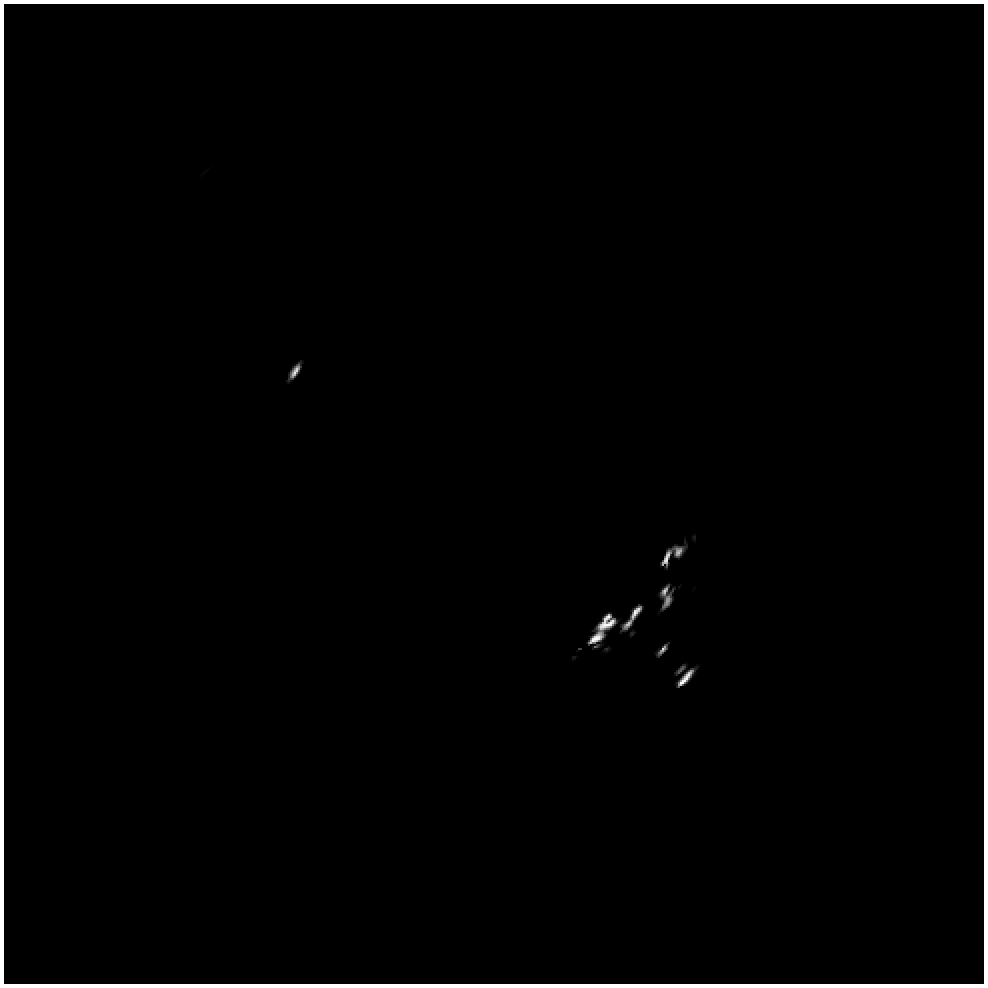

Figure 29. Radar image from Norfolk data set with almost no features.

Figure 30 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and RMS of the actual error (solid blue line). Due to the low power and range of the radar being used, the radar was not able to “see” very much land. This is why this data set shows the highest RMS error presented in this report, at 961m.

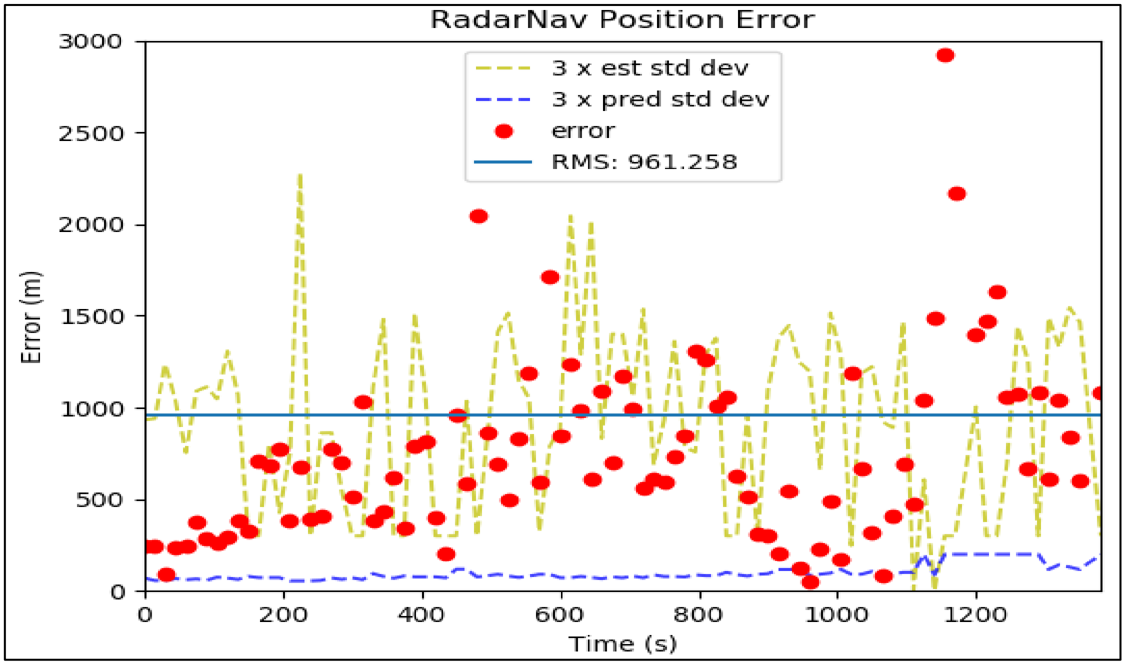

Figure 30. Plot of position errors vs. time for Norfolk, VA data set.

Figure 31 shows the same data as Figure 30 but in terms of pixels instead of distance. Not surprisingly, this data set also has the highest RMS pixel error in this report at 16 pixels (rounded from 15.9 because fractions of a pixel don't exist).

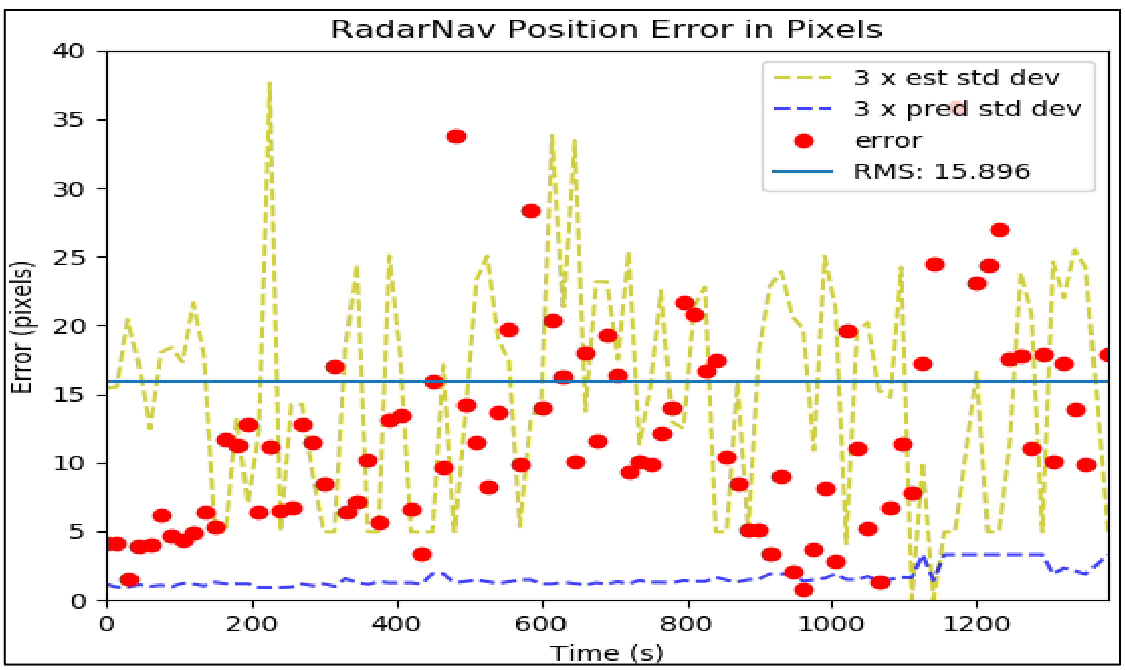

Figure 31. Plot of pixel errors vs. time for Norfolk, VA data set.

5.4 SWEDEN – RUN 1

The radar data from all of the Sweden datasets was collected for the purposes of post-processing. This data was collected using the collection hardware described in Section 2.2. The vessel hosting the radar was a Swedish Combat Boat 90 (CB90) which is a class of military fast assault craft used by the Swedish Navy. The data was collected by NIWC Pacific employees in conjunction with the testing of their own navigation system known as Advanced Scalable Assured PNT (ASAP); and with the assistance of the Swedish Defense Material Administration (FMV). The radar in use for this dataset was a Furuno DRS6A X-Band radar with a maximum power of 6Kw, and a maximum range of 120 nautical miles. The antenna in use was 4ft long. The exact locations used in the Sweden data sets could not be stated in this report. Therefore, the maps in Sections 5.4 - 0 have had the terrain and any reference to the latitudes and longitudes removed.

Figure 32 shows a blank map of the Stockholm Archipelago on the Baltic coast of Sweden, with the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). For the first run, the vessel started out at the North end of the track heading South, and moved counter-clockwise around several islands. The radar in use was more powerful than the one used for the Norfolk data set, but it had very good range (see Table 4). That, coupled with the fact that there was "visible" land on all sides of the radar, it is not surprising that the Sweden data sets have some of the lowest errors. In Figure 32 the estimated error is so low that it appears to be only a few sporadic ellipses rather than a large area completely surrounding the entire track, as it was for the other data sets. In reality, the estimated error ellipses do surround the entire track, but they are difficult to see due to the zoom level of the map and the fact that they are so much smaller than error ellipses in other data sets. The purple ellipses that are clearly visible in Figure 32 illustrate a unique issue that will need to be addressed in future versions of RadarNav, and it is due to the geographic features of this area. Basically, the radar was too close to land masses with relatively high elevations and was essentially "blinded" by them. The Stockholm Archipelago is dotted with thousands of land masses of various shapes, sizes and elevations. Many of these land masses slope very steeply into the water, allowing small and medium sized vessels to get fairly close to the land and pass through relatively narrow passages. Unfortunately for RadarNav, this temporarily occludes much of the surrounding land, making it difficult to match real and synthesized radar images. The good news is that the numerous land masses create many unique features to match between images, so it doesn't take RadarNav very long to recover after passing through a highly occluded area. The purple ellipses in Figure 32 show when the vessel was getting too close to land, or going through a narrow passage. They also show RadarNav's quick recovery.

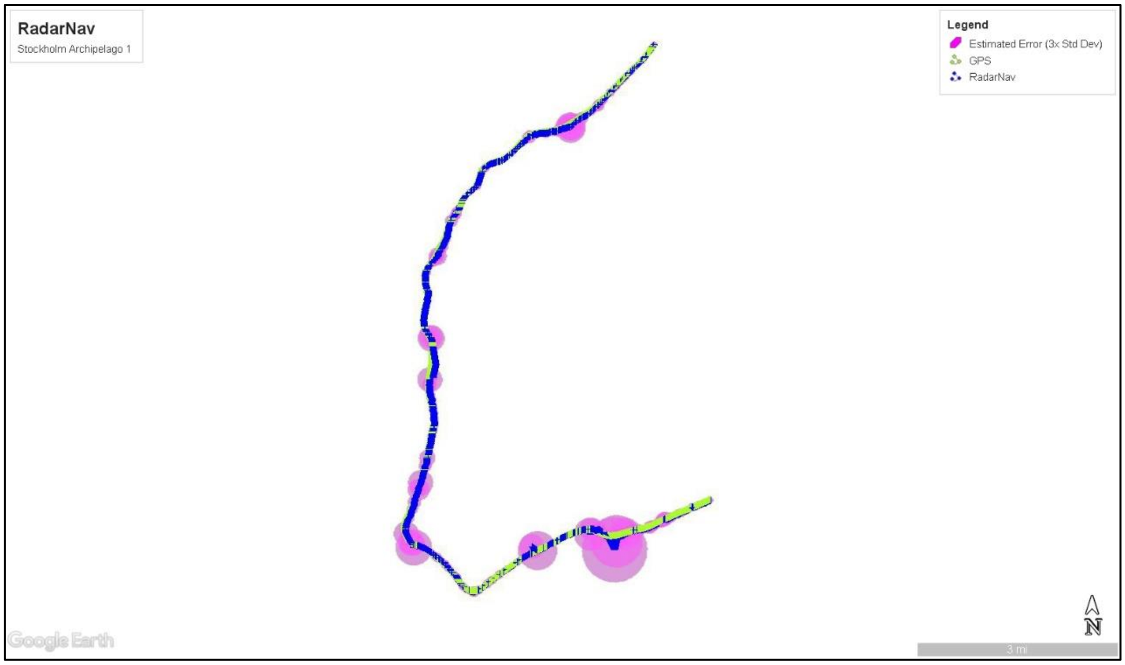

Figure 32:Map of RadarNav, GPS positions and estimated error for Sweden – Run 1.

Figure 33 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and RMS of the actual error (solid blue line). With an RMS error of only 35.5m, the first Sweden run has the lowest RMS error of all the data sets presented in this report. Most of the error values in Figure 33 fall below 35m, but the few outliers are large enough to raise the RMS value. As seen on the map in Figure 32, these outliers occur when the vessel passes through narrow passageways, or too close to land. Figure 33 illustrates that RadarNav was able to recover to its original error level as soon as the vessel moved further away from the land. Figure 33 also shows that the estimated error tracks the real error very well. Each of the data points for the larger error values falls on a spike that represents a larger estimated error. In an operational environment, these data points could be ignored to improve RadarNav’s position estimates.

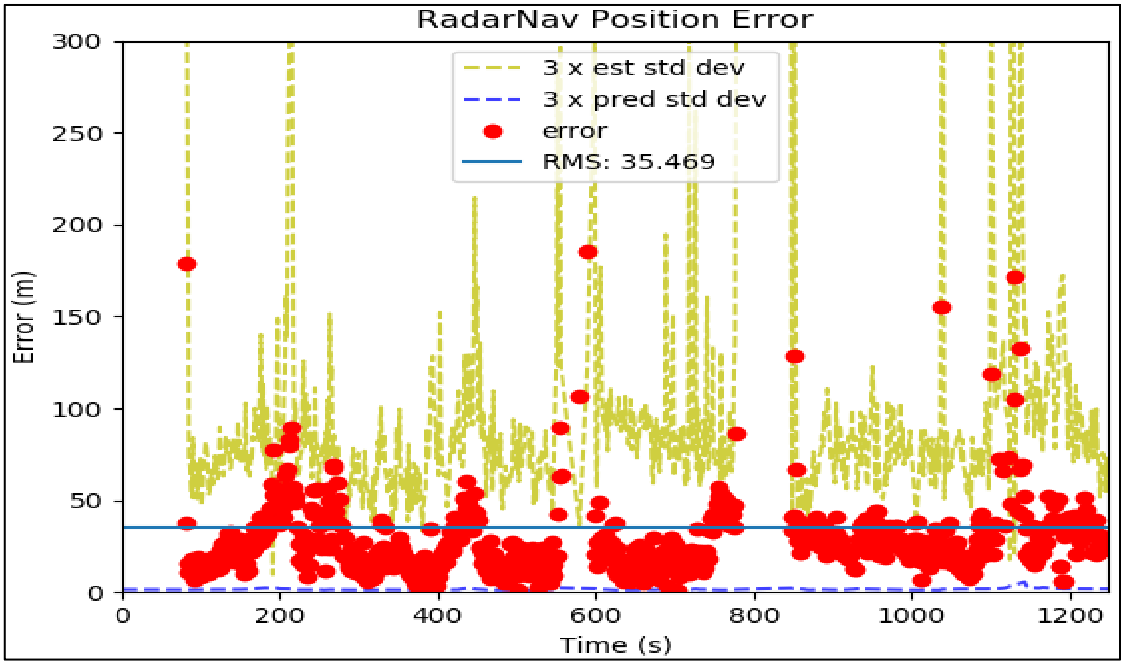

Figure 33. Plot of position errors vs. time for Sweden – Run 1 data set.

Figure 34 shows the same data as Figure 33 but in terms of pixels instead of distance. Even though this data set had the lowest RMS position error, the RMS pixel error of 3 is just a bit below average for the data sets presented in this report. In fact, before rounding up, this isn't even the lowest RMS pixel error among the four Sweden data sets. This shows that the overall error depends on more than just the resolution of the maps and radar images because these resolutions were the same for all of the Sweden data sets.

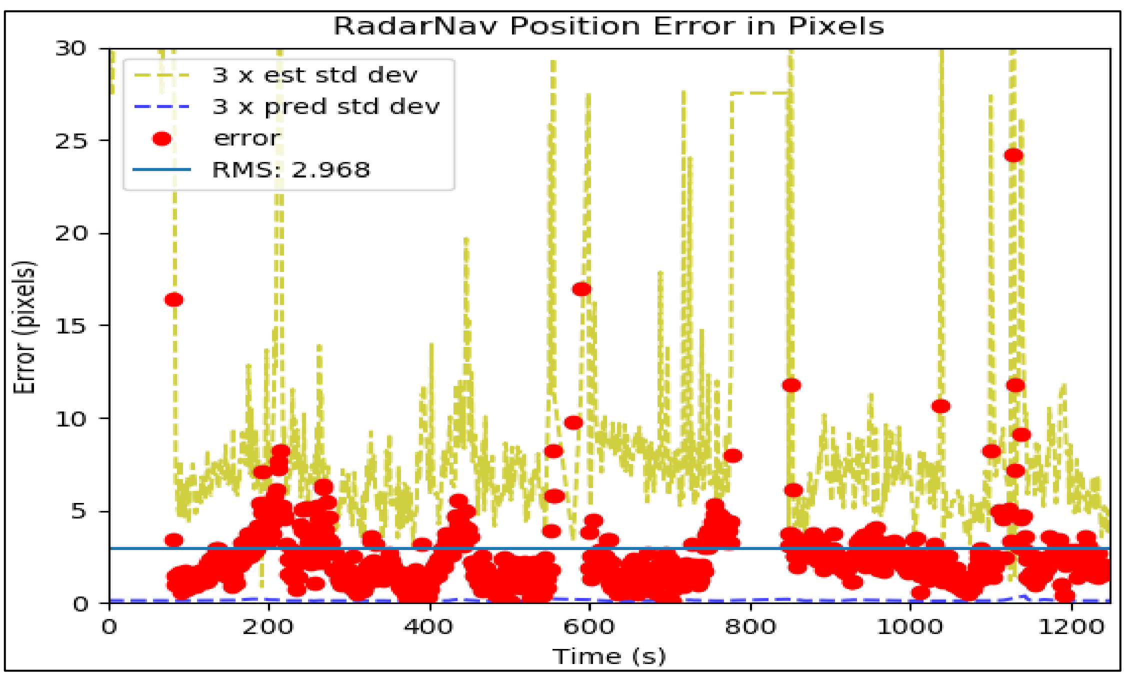

Figure 34. Plot of pixel errors vs. time for Sweden – Run 1 data set.

5.5 SWEDEN – RUN 2

The radar data from all of the Sweden datasets was collected in the same manner, and using the same hardware as described in Section 5.4 for the Sweden – Run 1 dataset. Figure 35 shows a blank map of the Stockholm Archipelago on the Baltic coast of Sweden, with the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). For the second run, the vessel started North of the track, and headed South through the islands. Just like the first run through the Stockholm Archipelago, RadarNav's estimated and actual errors were fairly low and only became a problem when passing close to land or through narrow passages.

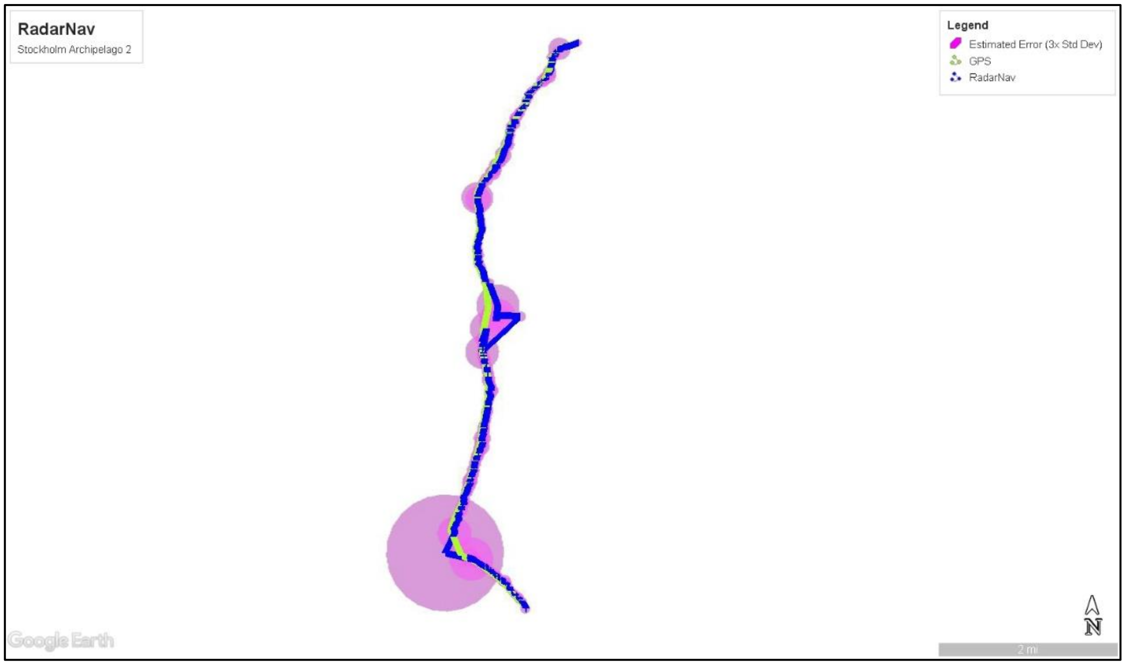

Figure 35. Map of RadarNav, GPS positions and estimated error for Sweden – Run 2.

Figure 36 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and RMS of the actual error (solid blue line). The route the vessel took for Sweden-Run 2 is the same route taken during the middle section of the Sweden-Run 1. Although the paths were very similar, the RMS error of Sweden-Run 1 was about 22m lower that Sweden-Run 2. From Figure 36 it is evident that this higher RMS error was primarily caused by two large spikes in the error. On The map in Figure 35 these two error spikes are seen about midway through the track where the vessel passed very close to land, and near the end of the track. As with the previous data sets, the estimated error tracks with the actual error; even with these large spikes.

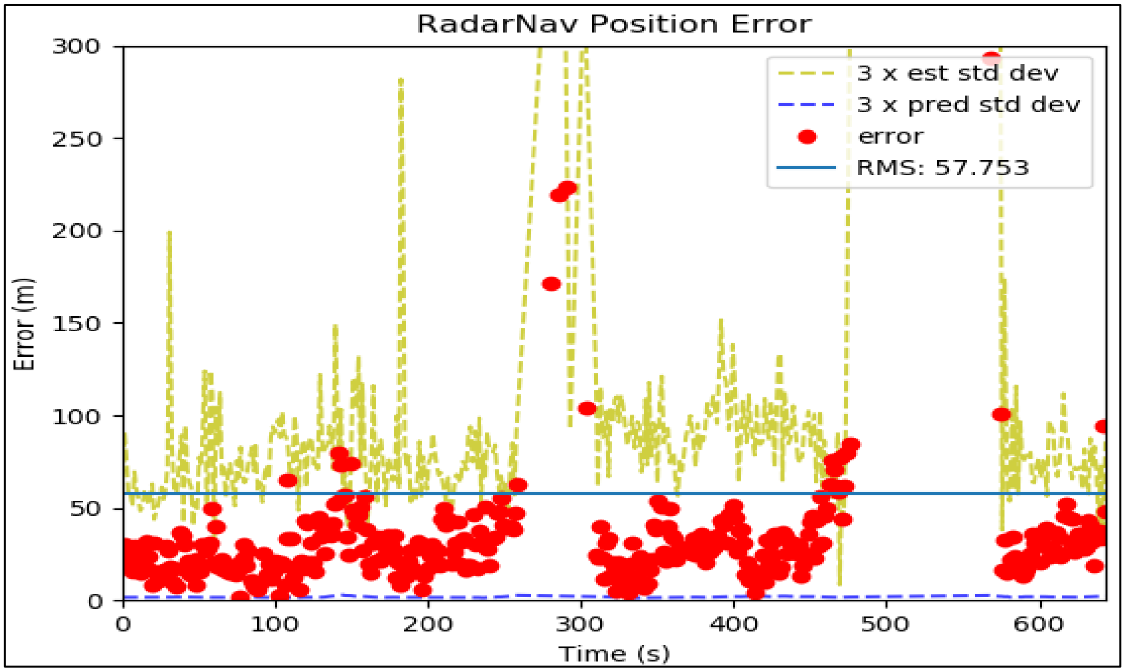

Figure 36. Plot of position errors vs. time for Sweden – Run 2 data set.

Figure 37 shows the same data as Figure 36 but in terms of pixels instead of distance. The RMS pixel error of 4 pixels is higher than the first Sweden run. This can also be attributed to the large error spikes mentioned previously.

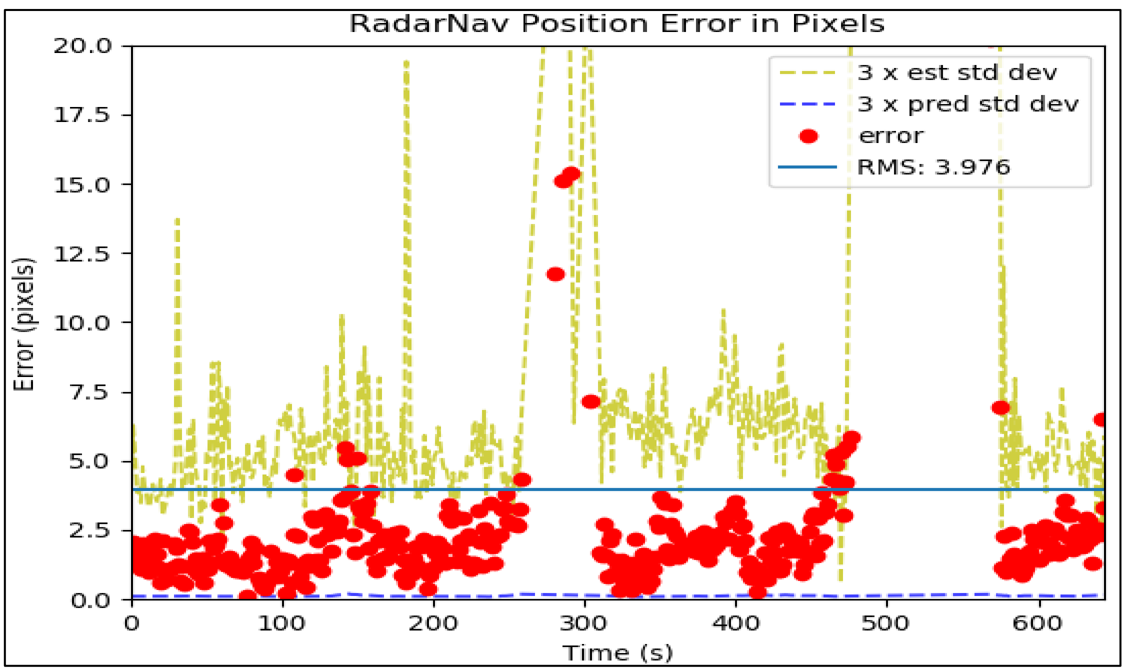

Figure 37. Plot of pixel errors vs. time for Sweden – Run 2 data set.

5.6 SWEDEN – RUN 3

The radar data from all of the Sweden datasets was collected in the same manner, and using the same hardware as described in Section 5.4 for the Sweden – Run 1 dataset. Figure 38 shows a blank map of the Stockholm Archipelago on the Baltic coast of Sweden, with the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). For the third run, the vessel started in the South, and headed North along the Eastern edge of a grouping of islands, with a relatively large distance to the closest islands to the West of the track. Just like the previous runs through the Stockholm Archipelago, RadarNav's estimated and actual errors were fairly low but RadarNav struggled when the vessel passed close to land or through narrow passages. This run had the highest errors of all the Sweden data sets. This is may be due to the fact that the vessel traveled along the Eastern edge of the grouping of islands rather than in between them so that the land features to the West of the radar were further away than the land to the East. This coincides with the larger estimated error ellipses about ¾ of the way through the track in Figure 38.

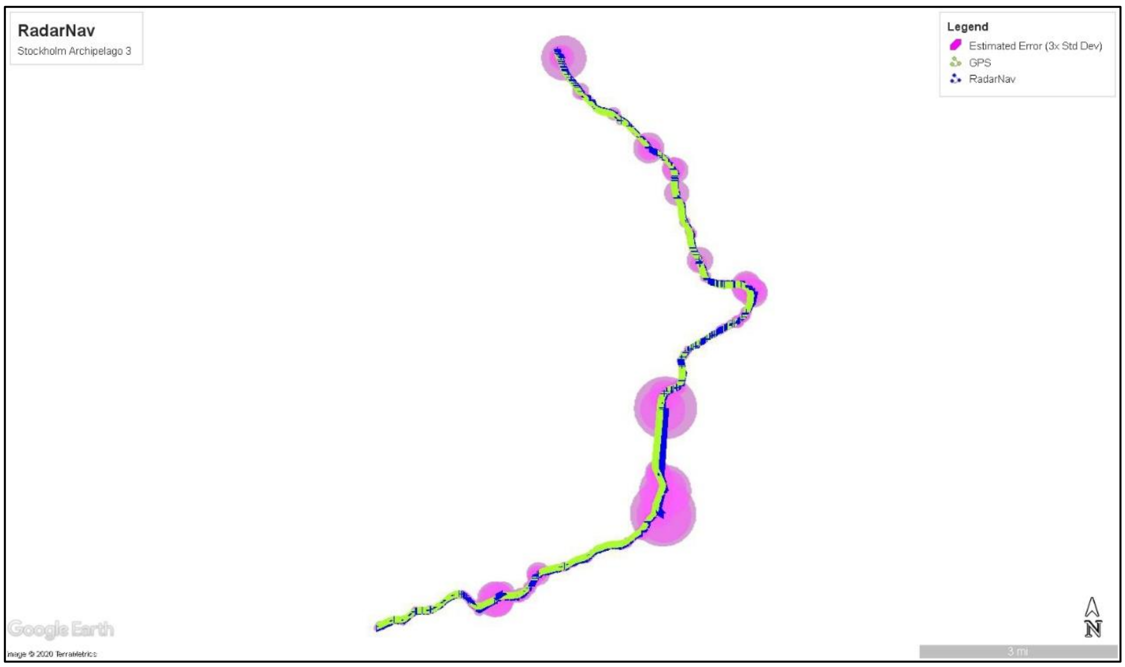

Figure 38. Map of RadarNav, GPS positions and estimated error for Sweden – Run 3.

Figure 39 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and RMS of the actual error (solid blue line). In addition to having the largest RMS position error of all the Sweden data sets, its baseline position error is also higher than the first two Sweden runs. This offset is also seen in other data sets where all the land is on one side of the radar. What is interesting about Figure 39 is that the large spikes in the estimated and actual errors (just after 400 seconds) are tracked by a spike in the predicted error. This actually happens with the other data sets as well, but the predicted error is so low that the variations can be difficult to see in these plots, and usually appear as just flat lines around zero.

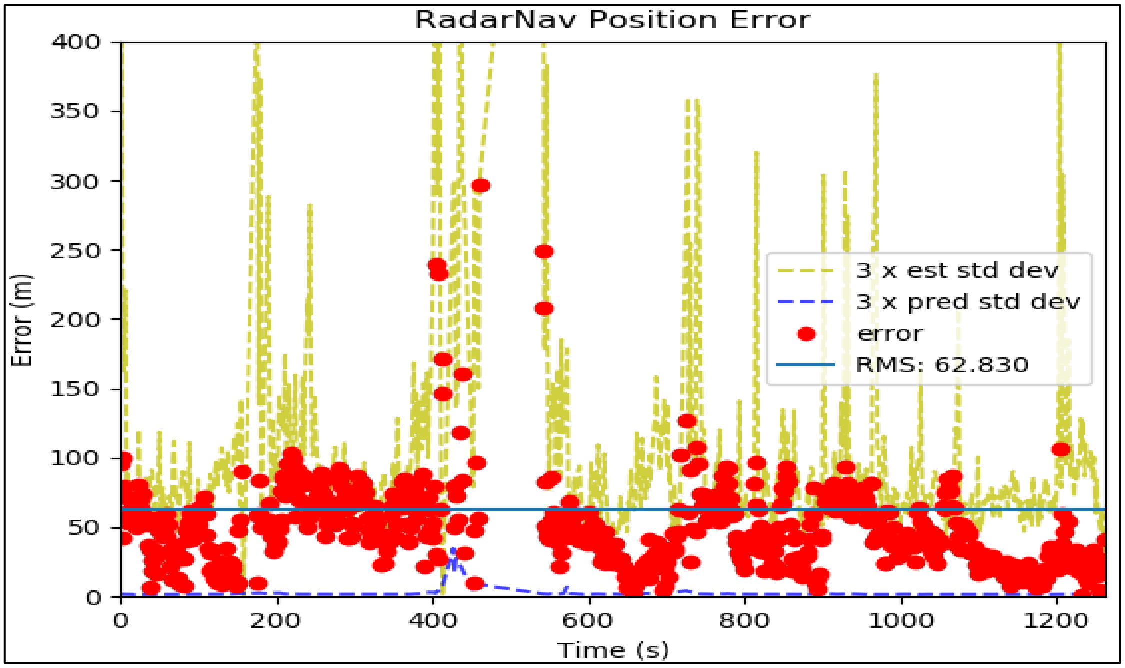

Figure 39. Plot of position errors vs. time for Sweden – Run 3 data set.

Figure 40 shows the same data as Figure 39 but in terms of pixels instead of distance.

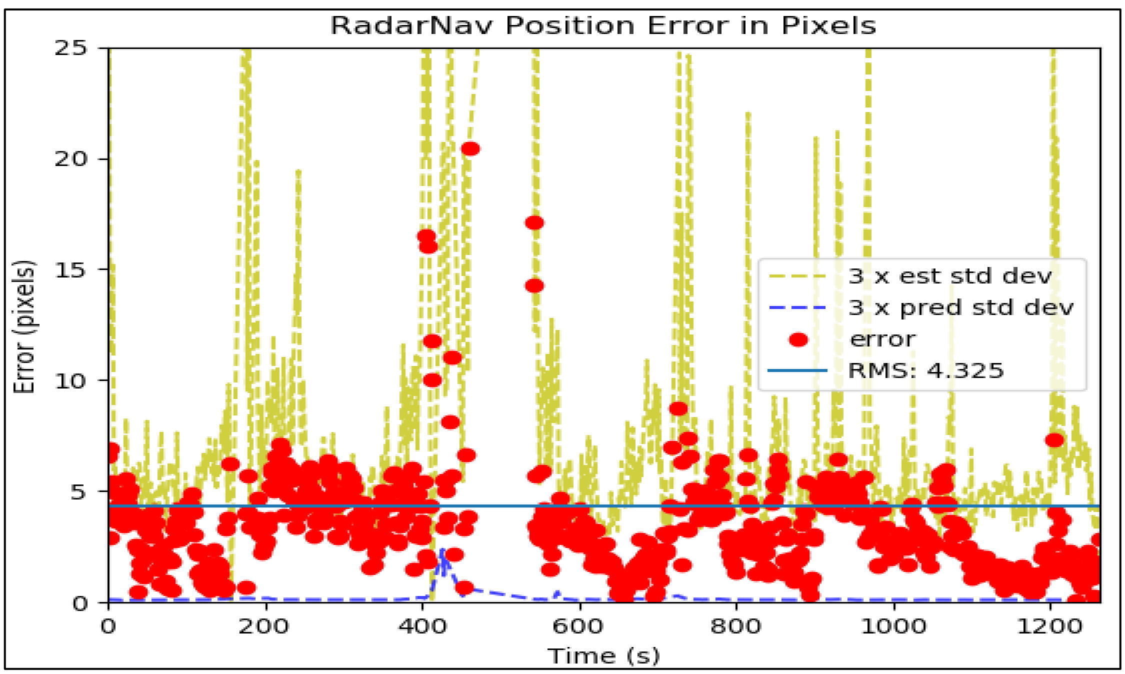

Figure 40. Plot of pixel errors vs. time for Sweden – Run 3 data set.

5.7 SWEDEN – RUN 4

The radar data from all of the Sweden datasets was collected in the same manner, and using the same hardware as described in Section 5.4 for the Sweden – Run 1 dataset. Figure 41 shows a blank map of the Stockholm Archipelago on the Baltic coast of Sweden, with the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). For the fourth run, the vessel started in the North, and headed South through the islands. Just like the previous runs through the Stockholm Archipelago, RadarNav's estimated and actual errors were fairly low but RadarNav struggled when the vessel passed close to land or through narrow passages.

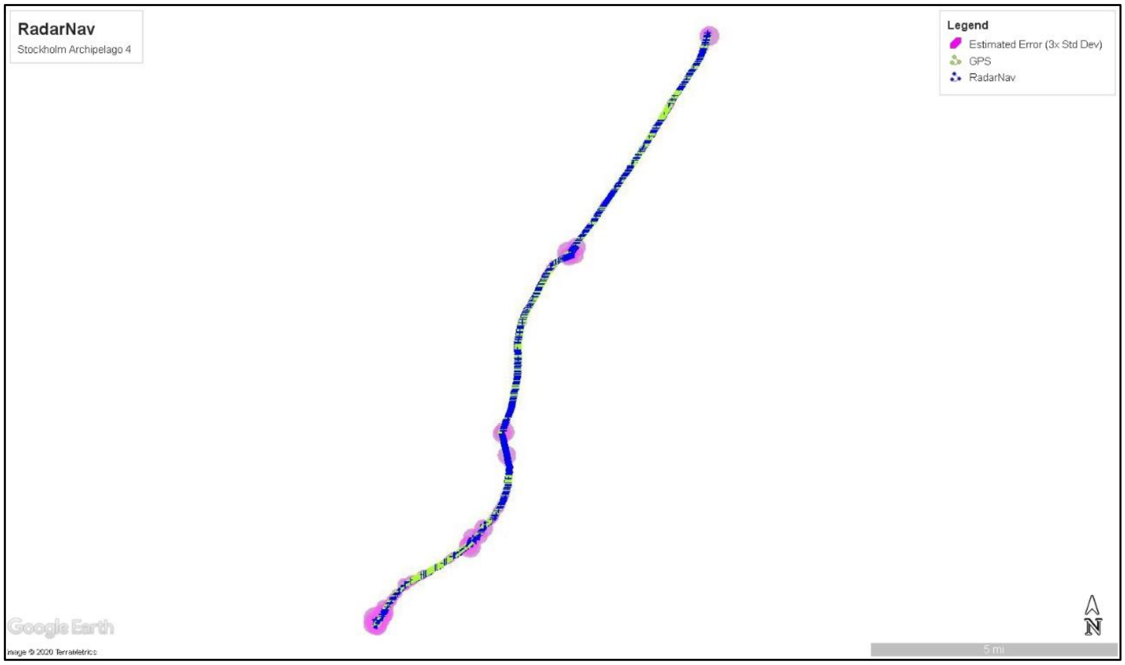

Figure 41. Map of RadarNav, GPS positions and estimated error for Sweden – Run 4.

Figure 42 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and RMS of the actual error (solid blue line). This plot looks very similar to plots from the previous Sweden data sets. The error is fairly consistent with spikes when the vessel passed very close to land, or through narrow passageways.

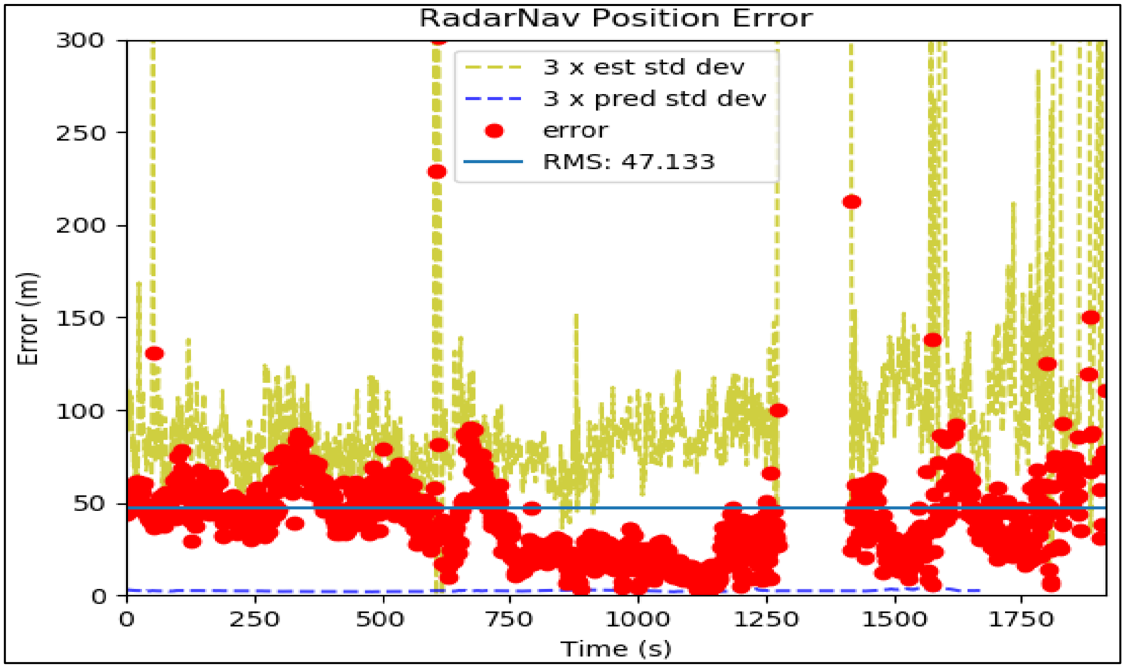

Figure 42. Plot of position errors vs. time for Sweden – Run 4 data set.

Figure 43 shows the same data as Figure 42 but in terms of pixels instead of distance.

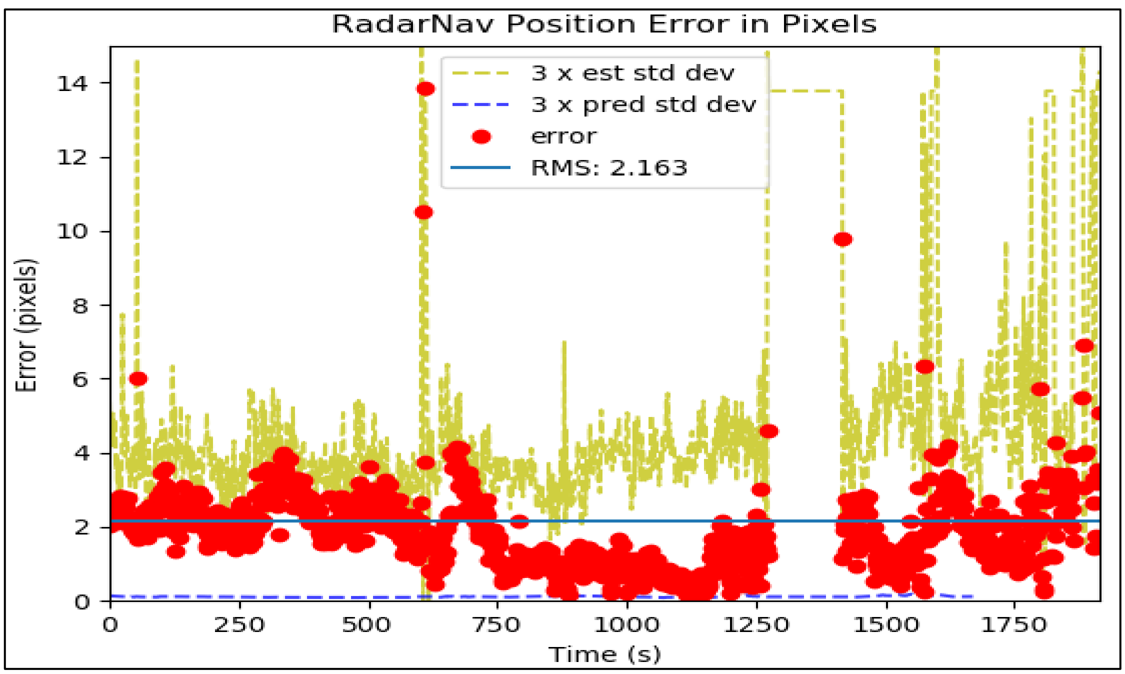

Figure 43. Plot of pixel errors vs. time for Sweden – Run 4 data set.

5.8 ASTORIA, OREGON - COLUMBIA RIVER DELTA

The radar data from the Astoria dataset was collected for the purposes of post-processing. This data was collected using the collection hardware described in Section 2.2. The vessel hosting the radar was the R/V Kilo Moana, which is part of the University-National Oceanographic Laboratory System (UNOLS), and operates out of the University of Hawaii Marine Center in Oahu, Hawaii. The R/V Kilo Moana is a 186’ Small Waterplane Area Twin Hull (SWATH) vessel. More information on the Kilo Moana is available on her Wikipedia page at the link in (Wikipedia, n.d.). The radar in use for this dataset was a Furuno FAR-3230 S-Band radar with a maximum power of 30Kw, and a maximum range of 96 nautical miles. The antenna in use was 12ft long. The rest of the radar systems used to collect data are all X-Band (10GHz) radars. However, this one is an S-Band (3GHz) radar. X-Band radars provide a sharper image and better resolution, but S-Band radars are especially useful for operating in heavy rain and fog. They are also better for long range bird detection. S-Band radar also requires larger antennas.

Figure 44 shows a map of West coast of Oregon with the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). This data was supposed to include the vessel’s transit out of the Columbia River delta on the way back to Hawaii. Unfortunately, the radar capture hardware was not turned on until the vessel was about 40Km from the mouth of the river. Nevertheless, this data set illustrates the importance of looking at the pixel error when validating the RadarNav algorithm. It also shows why future versions of RadarNav should have the ability to work with increased numbers of pixels. In this data set, the vessel was about 30Km away from the coast. The radar was still able to see the coastline, but the meter per pixel (m/p) resolution needed to be increased in order to fit the entire radius of the radar’s range onto a single map. This data set is also important because it shows that RadarNav still functions at longer distances from the coast. The real limiting factor is how much land the radar can “see” and how many unique features exist in each radar image that can be matched to features in the synthesized images.

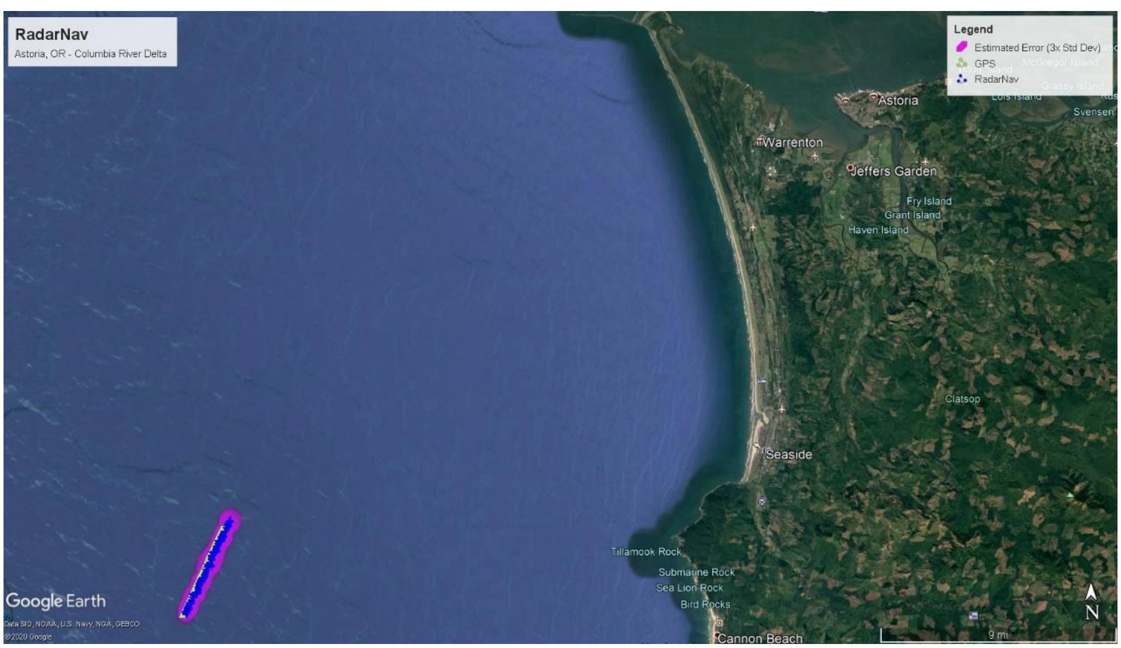

Figure 44. Map of RadarNav, GPS positions & estimated error for Astoria, OR.

Figure 45 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and RMS of the actual error (solid blue line). The RMS error is 389m which is pretty high compared to the previous data sets. The Norfolk data set had a higher RMS error, but that was due to the radar not being powerful enough to “see” enough land for properly matching to synthesized radar images. The radar system used to collect this data set is the most powerful of all the radars used to collect the data presented in this report. However, the vessel was about 28Km from land at its closes point in this data set, which is the largest distance presented in this report.

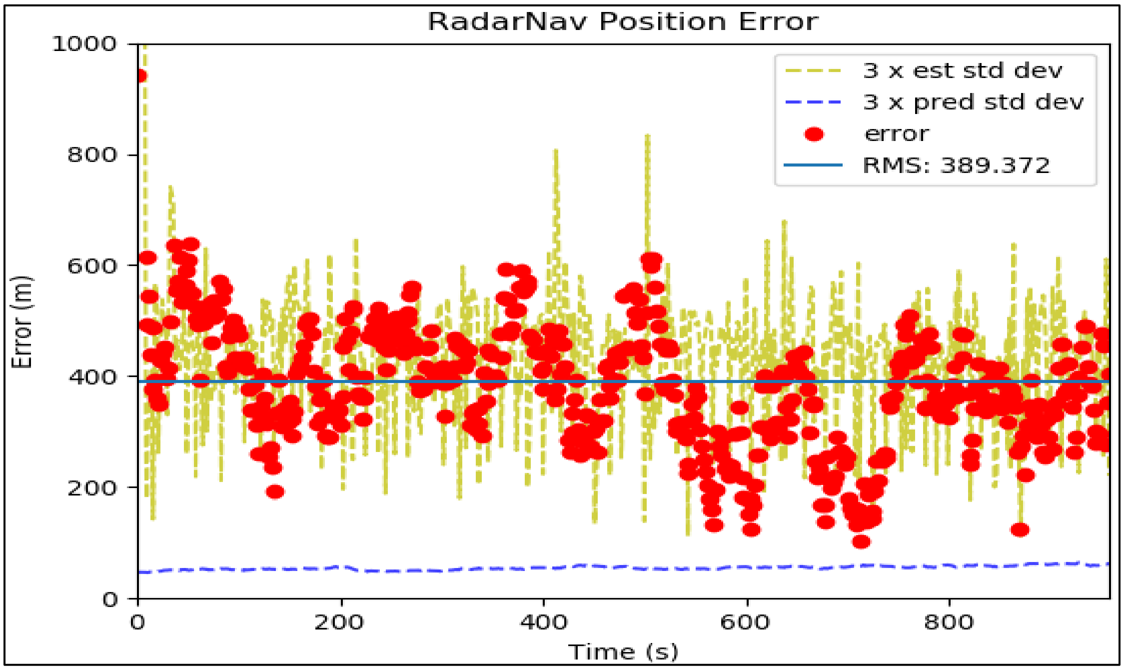

Figure 45. Plot of position errors vs. time for Astoria, OR data set.

Since the number of pixels is fixed for each image that RadarNav processes, the m/p resolution must be increased as the radar’s range increases. In order to fit the vessel and the land into the same image, the resolution was set to 188m/p for this data set. Figure 46 shows the same data as Figure 45 but in terms of pixels instead of distance. It also shows that the RMS pixel error for this data set was 3 pixels (rounded up) which is the same pixel error as several other data sets, and just below the average pixel error for all the data sets presented in this report. The large error in the Norfolk data set was due to the lack of land or unique features to match between the radar and synthesized images. With this Astoria data set, RadarNav was able to match images just fine, but the error that did exist was magnified by the large distance covered by each pixel. If RadarNav was set up to handle more pixels and each pixel covered a smaller distance, the position error for the Astoria data set would be smaller.

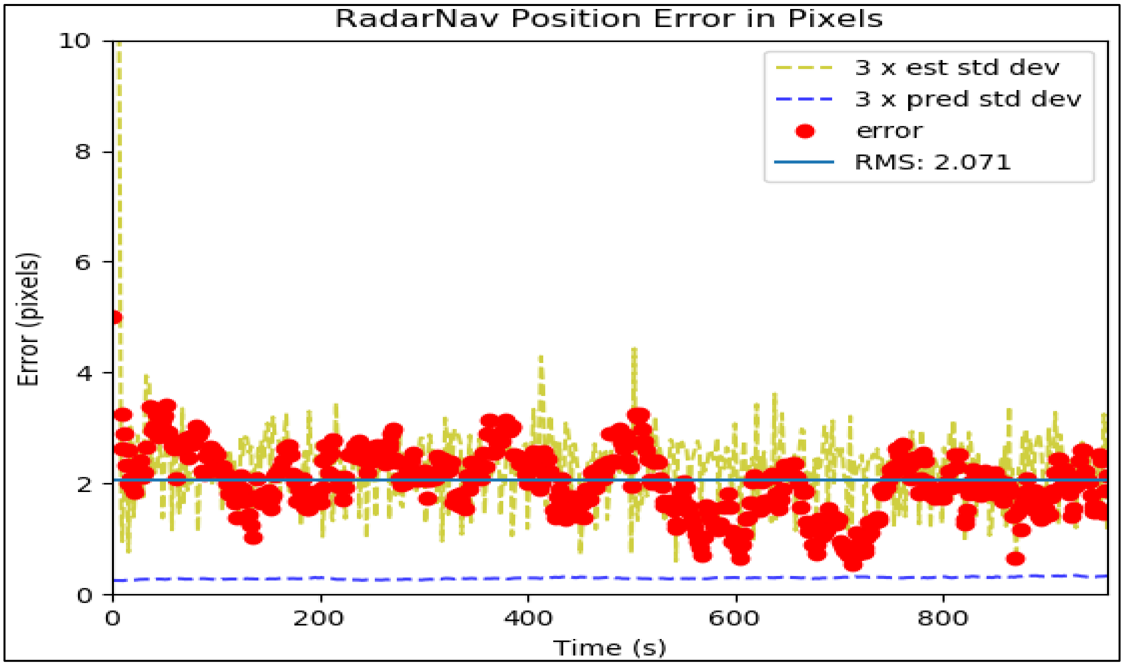

Figure 46. Plot of pixel errors vs. time for Astoria, OR data set.

5.9 HONOLULU, HAWAII – RUN 1

The radar data from all of the Hawaii datasets was collected in the same manner, and using the same hardware as described in Section 0 for the Astoria dataset. Figure 47 shows a map of the Island of O‘ahu with the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). For the first Honolulu run, the vessel started to the South-West of O‘ahu, then transited North along the island’s West coast. The beginning of this data set (southern end) the estimated error is higher than expected. This is most likely due to the fact that the vessel was holding station for nearly an hour at the beginning of this data set. During this time the vessel’s orientation was changing, but the distance traveled in any particular direction was small. Once the vessel began transiting North, the actual and estimated errors both dropped. As the vessel passed the sharp Ka'Ena Point, the sudden loss of land to the East caused the estimated error to begin growing. As the vessel continued to get further away from land, the error continued to grow. However, the actual position error stayed relatively low for a while past the point. By the end of this data set the closest land is over 17Km away. Since this same radar system was used to collect the Astoria data set, it should have been able to see plenty of land at this distance. However, the power and range of most radar systems can be changed by the user depending on the situation. In this instance, the radar was shorter and the meters per pixel resolution was smaller than with the Astoria data set.

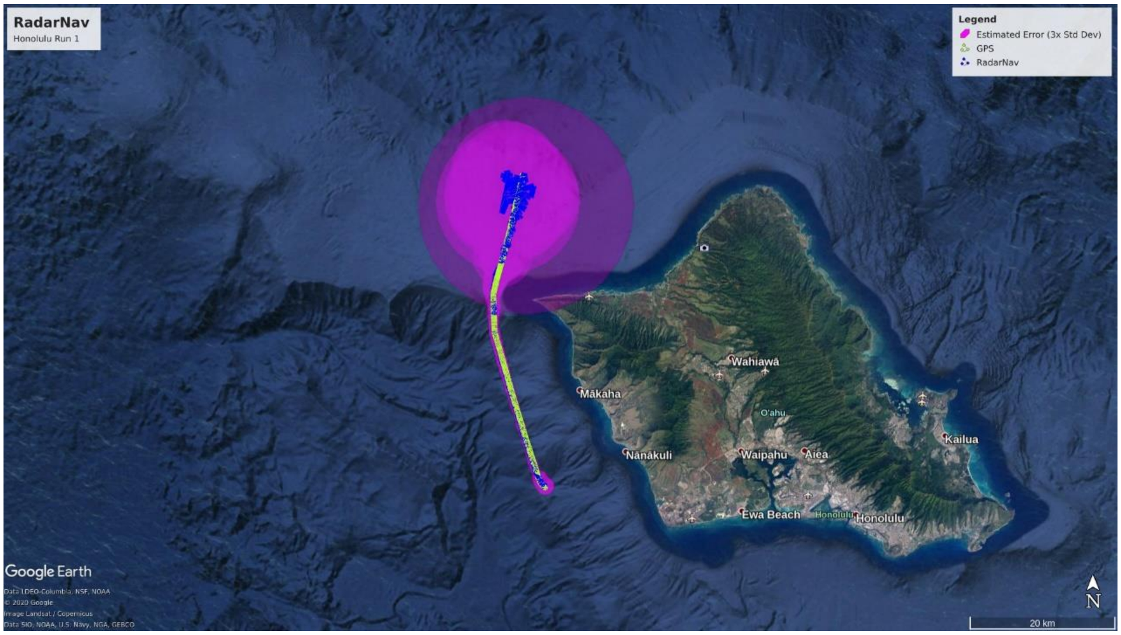

Figure 47. Map of RadarNav, GPS positions & estimated error for Honolulu, HI – Run 1.

Figure 48 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and RMS of the actual error (solid blue line). The estimated error start to grow around 9500 seconds (when the vessel passed the point), but the actual error did not really start growing for about 15 minutes.

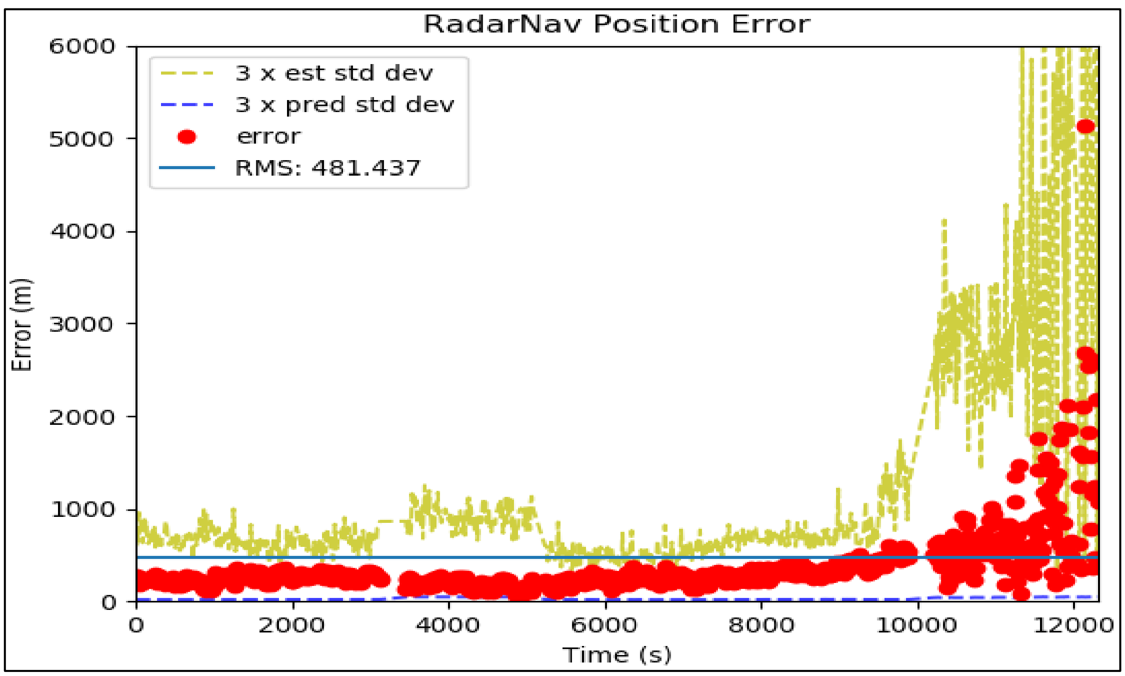

Figure 48. Plot of position errors vs. time for Honolulu, HI – Run 1 data set.

Figure 49 shows the same data as Figure 48 but in terms of pixels instead of distance.

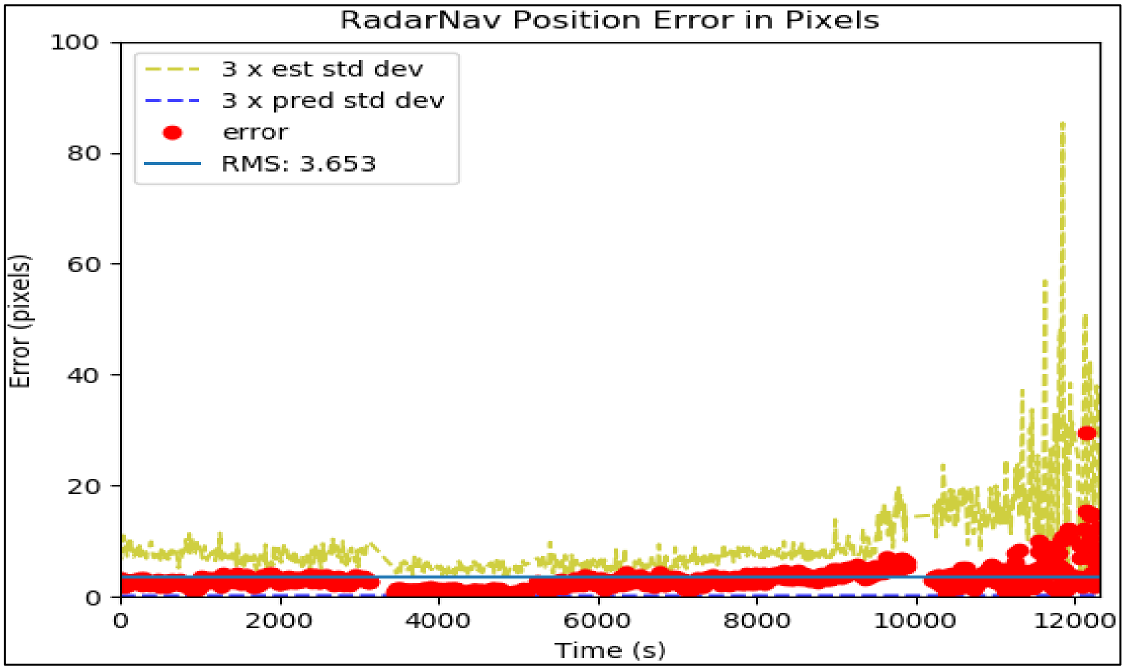

Figure 49. Plot of pixel errors vs. time for Honolulu, HI – Run 1 data set.

5.10 HONOLULU HAWAII – RUN 2

The radar data from all of the Hawaii datasets was collected in the same manner, and using the same hardware as described in Section 0 for the Astoria dataset. Figure 50 shows a map of the South-East corner of the Island of O‘ahu with the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). During the second Honolulu run, the vessel started about 7Km south of O‘ahu and traveled West until it passed the edge of the island. Then it turned to a North-West heading for the duration of the data set. Like the first Honolulu run, the vessel was keeping station at the beginning of this data set. The majority of this was removed from the plots so that the rest of the data was easier to see. There is a large jump in error about ¾ of the way through this data set. This is due to a change in the radar's range, which resulted in a short period of time where there was not enough land mass in the radar image to allow RadarNav to produce a solution. Once the range was increased and enough land mass was visible in the radar image, RadarNav was able to provide better position estimates. Fortunately, RadarNav was able to recover to its original error, and the estimated error also jumped up at the same time. If the estimated error was being used to gauge the trust of the RadarNav position estimates, these erroneous data points would have been ignored.

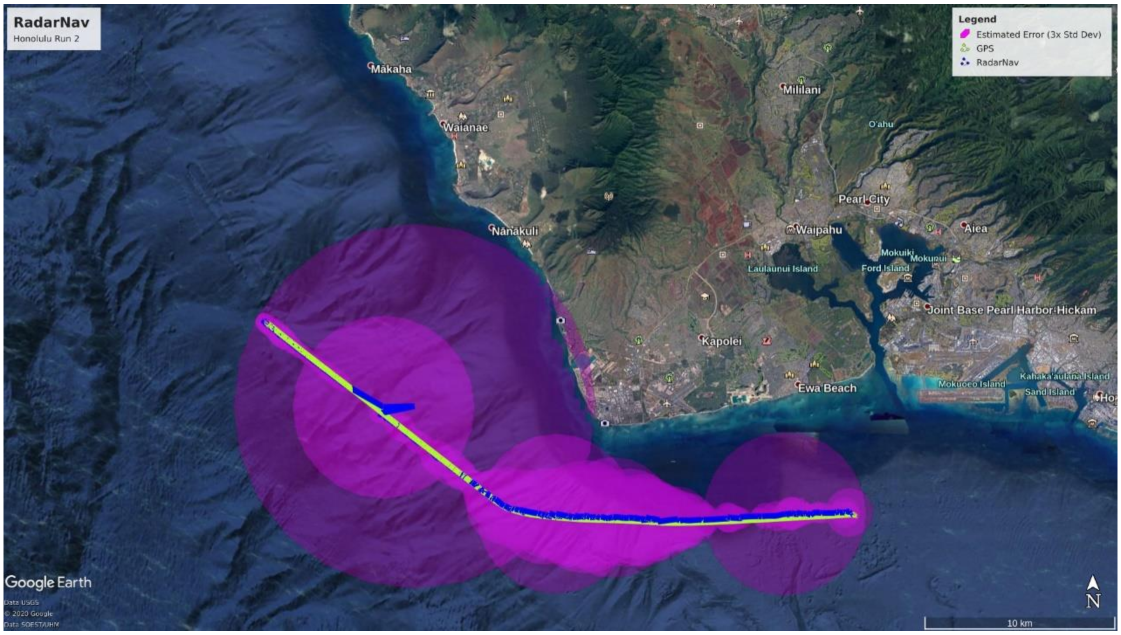

Figure 50. Map of RadarNav, GPS positions & estimated error for Honolulu, HI – Run 2.

Figure 51 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and RMS of the actual error (solid blue line). It is not currently known why the estimated error rises so much in the middle of this data set, while the actual error does not. The assumption is that the estimated error became higher as the vessel passed the edge of the island, but the actual error did not suffer as much as RadarNav expected it to.

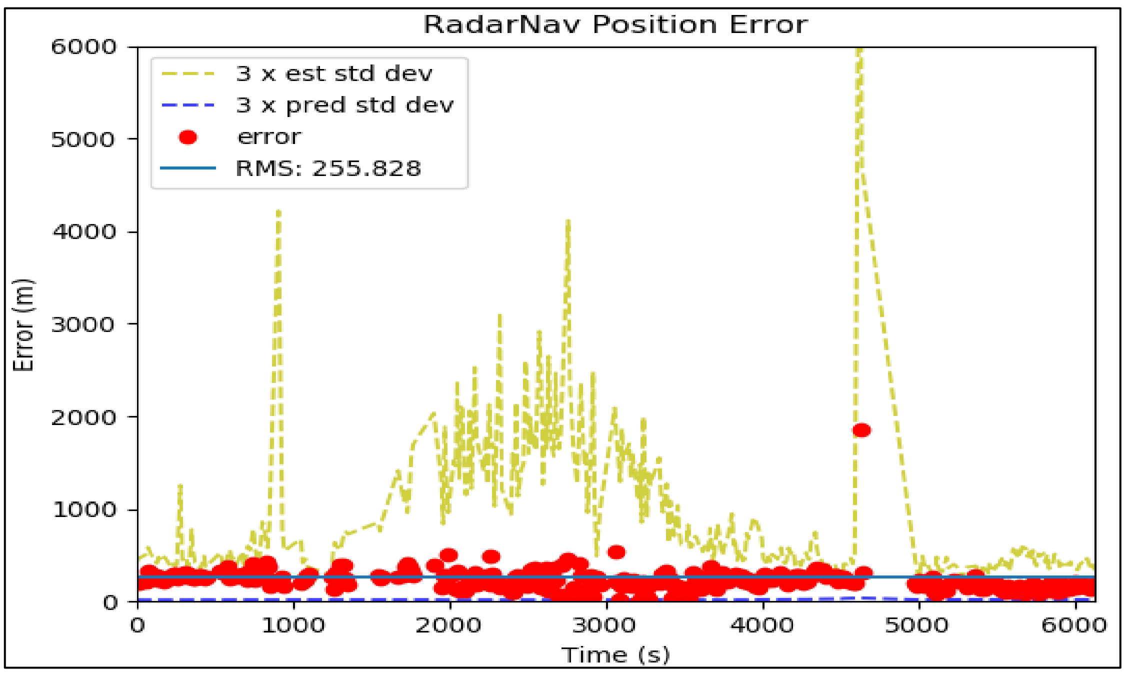

Figure 51. Plot of position errors vs. time for Honolulu, HI – Run 2 data set.

Figure 52 shows the same data as Figure 51 but in terms of pixels instead of distance.

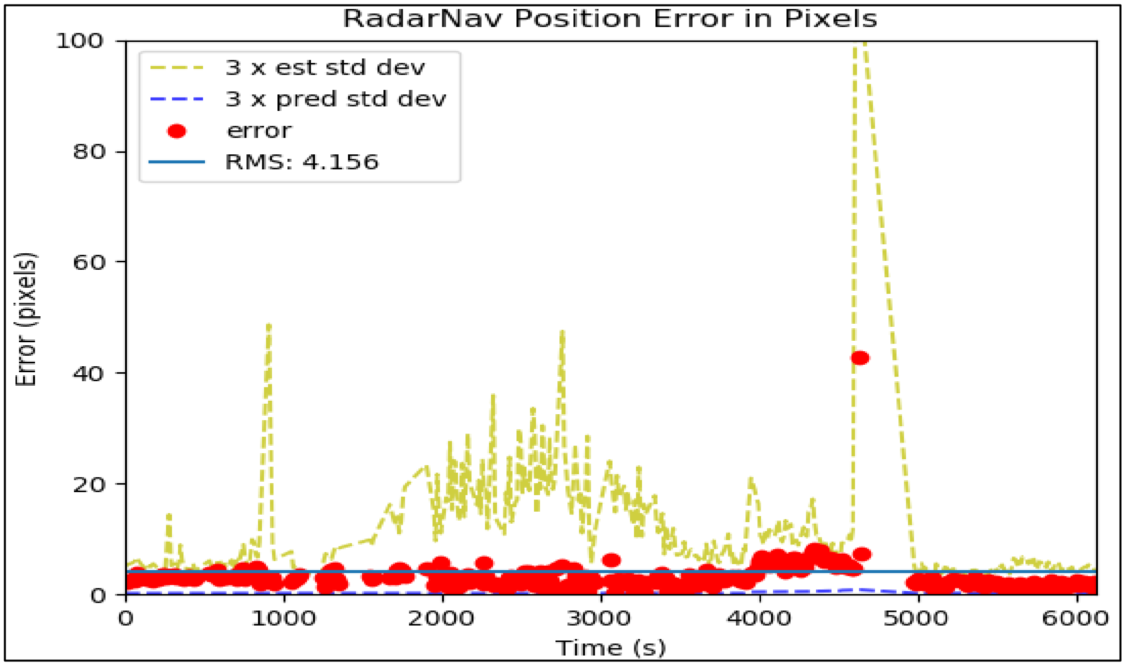

Figure 52. Plot of pixel errors vs. time for Honolulu, HI – Run 2 data set.

5.11 KAUA‘I HAWAII – RUN 1

The radar data from all of the Hawaii datasets was collected in the same manner, and using the same hardware as described in Section 0 for the Astoria dataset. Figure 53 shows a map of the Kaulakahi Channel that separates the islands of Ni‘ihau and Kaua‘i. The map also displays the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). The first Kaua‘i run starts off at the South end of the track, and the vessel transited in a nearly straight line on a North-West heading through the channel.

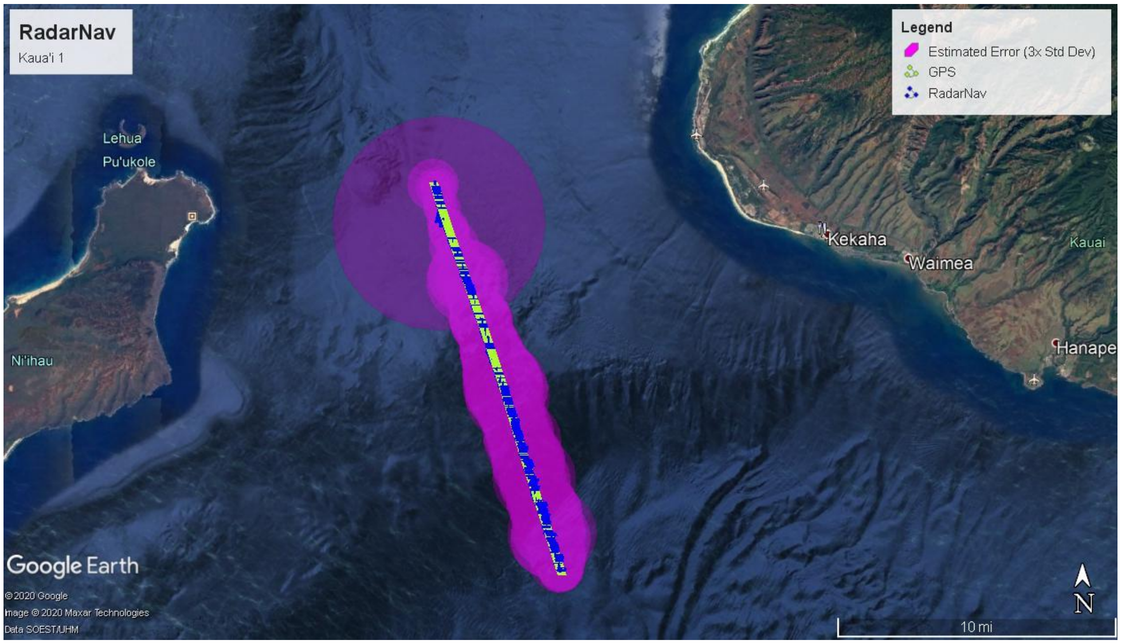

Figure 53. Map of RadarNav, GPS positions & estimated error for Kauaʻi, HI – Run 1.

Figure 54 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and RMS of the actual error (solid blue line). The large spike at the end of this data set was caused by a change in the radar's range. This was also the cause for the increase in error in the Honolulu-Run 2 data set.

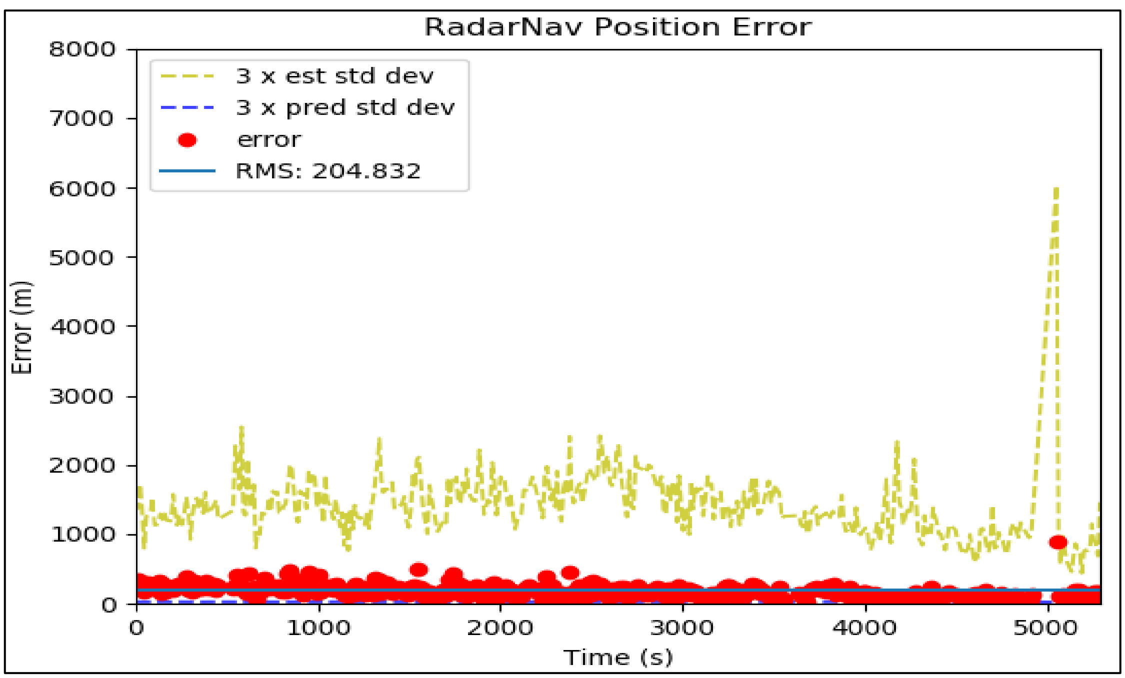

Figure 54. Plot of position errors vs. time for Kauaʻi, HI – Run 1 data set.

Figure 55 shows the same data as Figure 54 but in terms of pixels instead of distance.

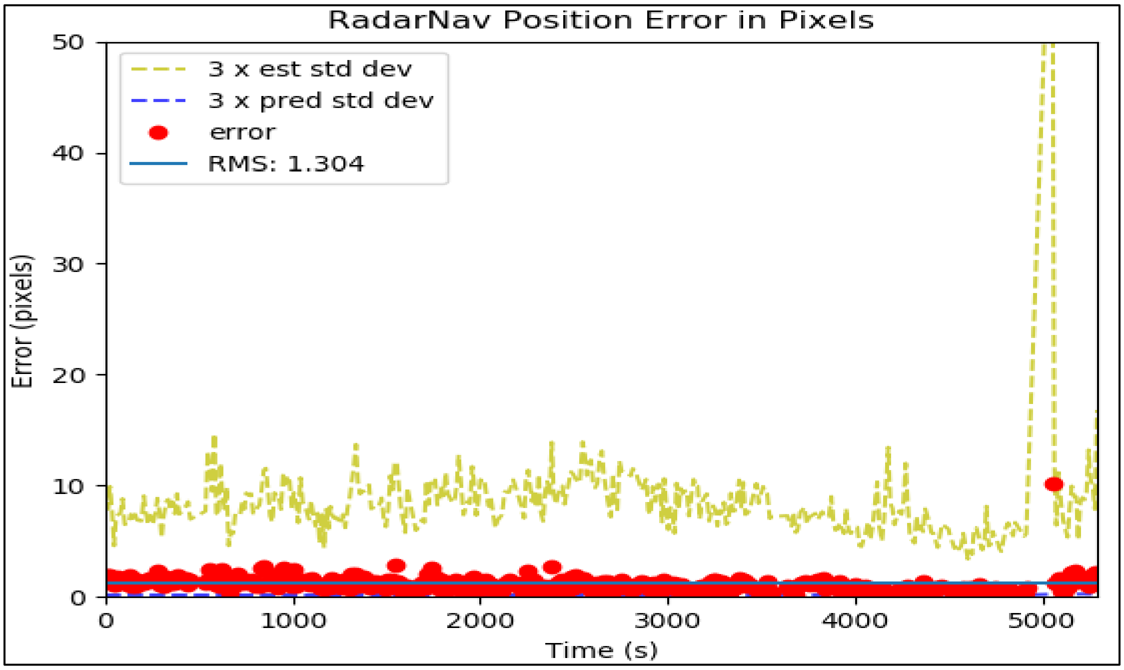

Figure 55. Plot of pixel errors vs. time for Kaua‘i, HI – Run 1 data set.

5.12 KAUA‘I HAWAII – RUN 2

The radar data from all of the Hawaii datasets was collected in the same manner, and using the same hardware as described in Section 0 for the Astoria dataset. Figure 56 shows a map of the Kaulakahi Channel that separates the islands of Ni‘ihau and Kaua‘i. The map also displays the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). Just like the first Kaua‘i run, the second run has the vessel starting off at the South end of the track and moving along a North-Western heading. With this data set, the vessel continuously moves farther away from land, but it is cut short just as the errors are starting to grow.

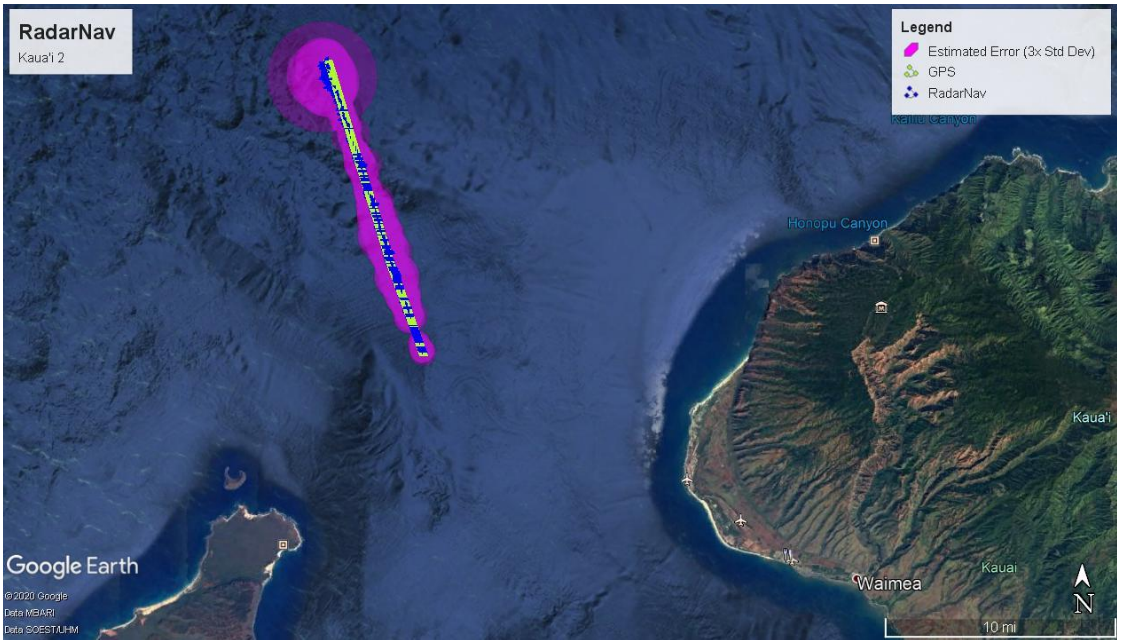

Figure 56. Map of RadarNav, GPS positions & estimated error for Kauaʻi, HI – Run 2.

Figure 58 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and RMS of the actual error (solid blue line).

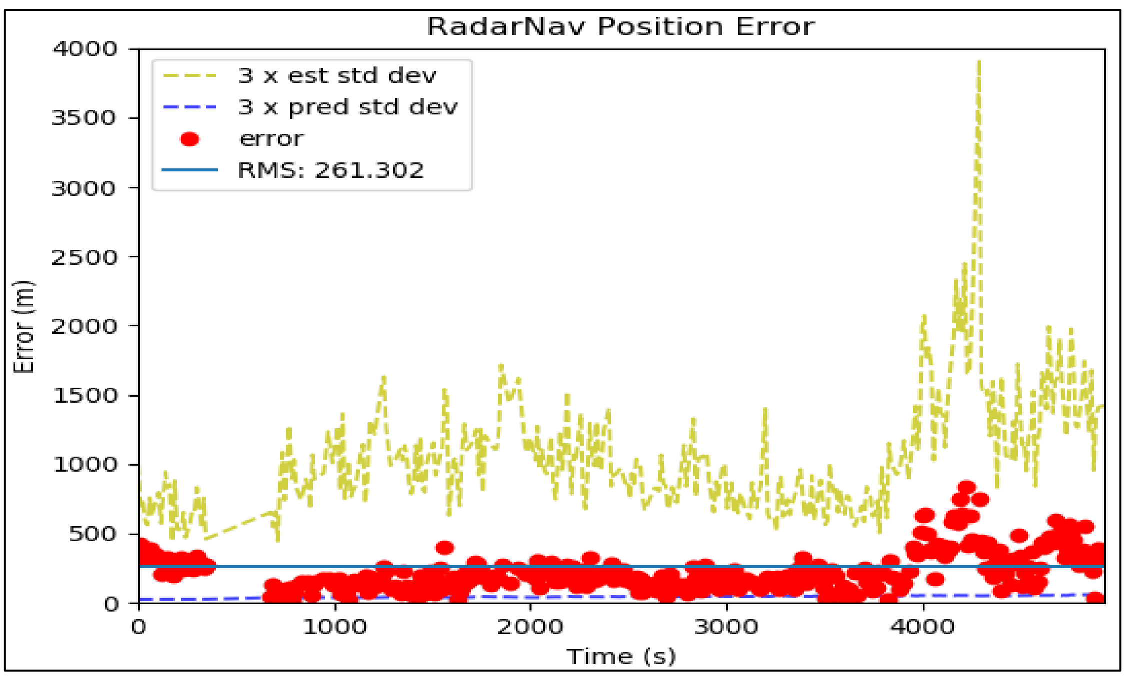

Figure 57. Plot of position errors vs. time for Kauaʻi, HI – Run 2 data set.

Figure 58 shows the same data as Figure 57 but in terms of pixels instead of distance.

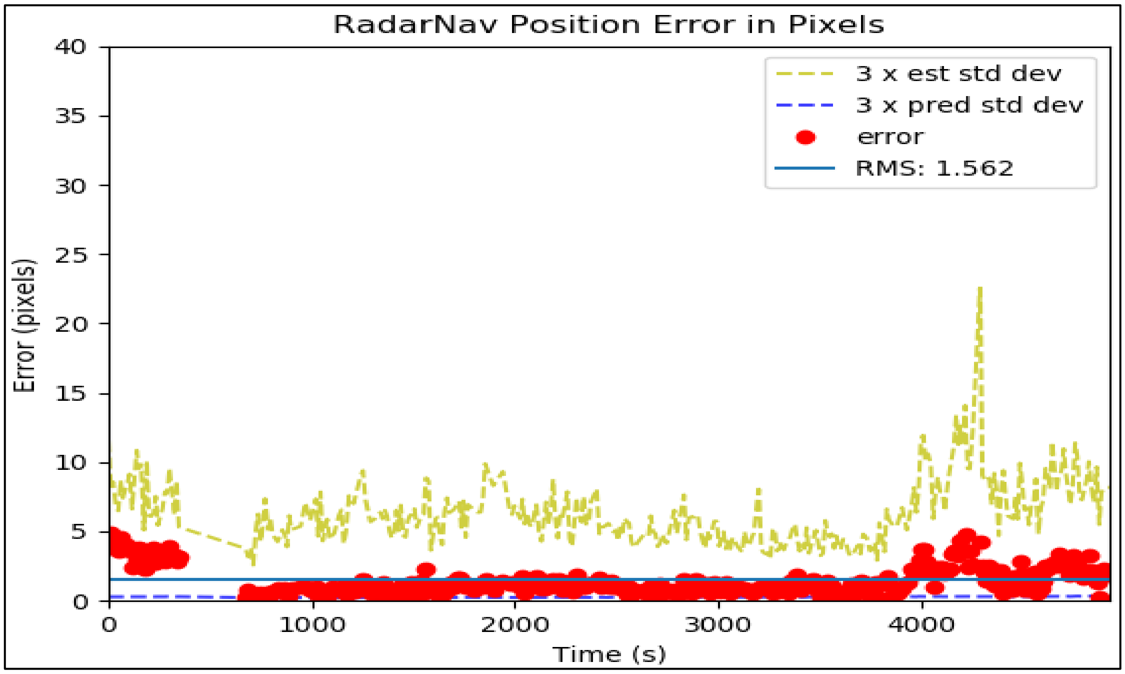

Figure 58. Plot of pixel errors vs. time for Kauaʻi, HI – Run 2 data set.

5.13 KAUAʻI HAWAII – RUN 3

The radar data from all of the Hawaii datasets was collected in the same manner, and using the same hardware as described in Section 0 for the Astoria dataset. Figure 59 shows a map of the Kaulakahi Channel that separates the islands of Niʻihau and Kauaʻi. The map also displays the RadarNav positions (blue track), GPS positions (green track) and the estimated error ellipses (purple area). For the third Kauaʻi run the vessel started from the North end of the track and transited along a South-East heading with a slight bend as it reached the westernmost edge of Kauaʻi.

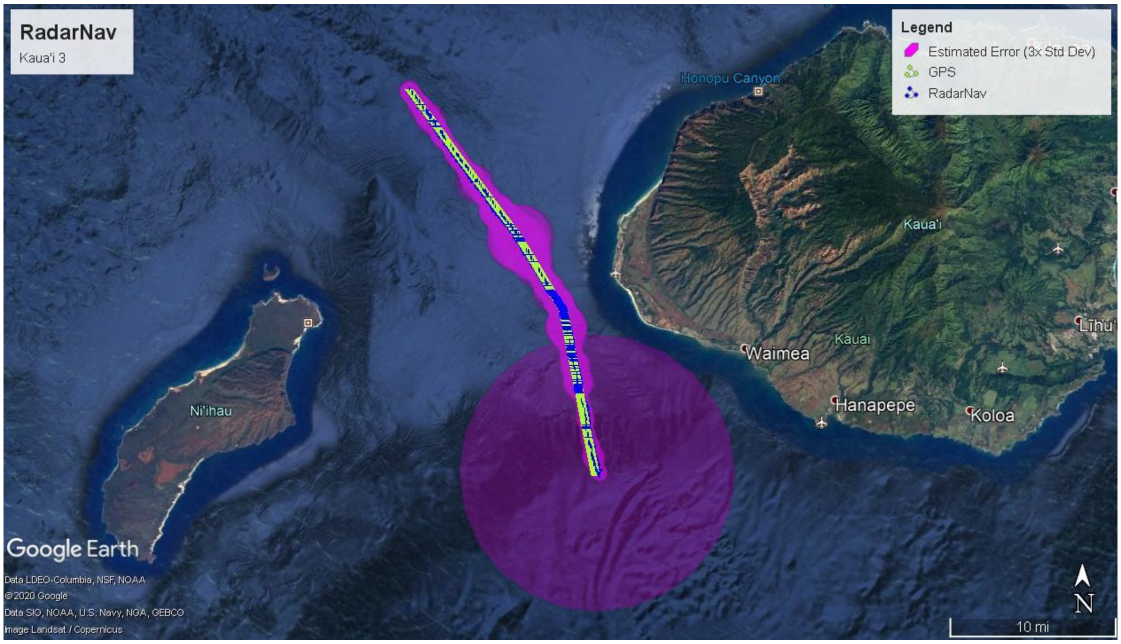

Figure 59. Map of RadarNav, GPS positions & estimated error for Kauaʻi, HI – Run 3.

Figure 60 shows a plot of the estimated error (yellow dashed line), predicted error (dashed blue line), actual error (red dots) and RMS of the actual error (solid blue line). There are a couple jumps in the estimated error that coincide with jumps in the actual error. These are most likely due to noise similar to what was discussed in Section 2.2. There were also several other large ships in the area that would add to this noise.

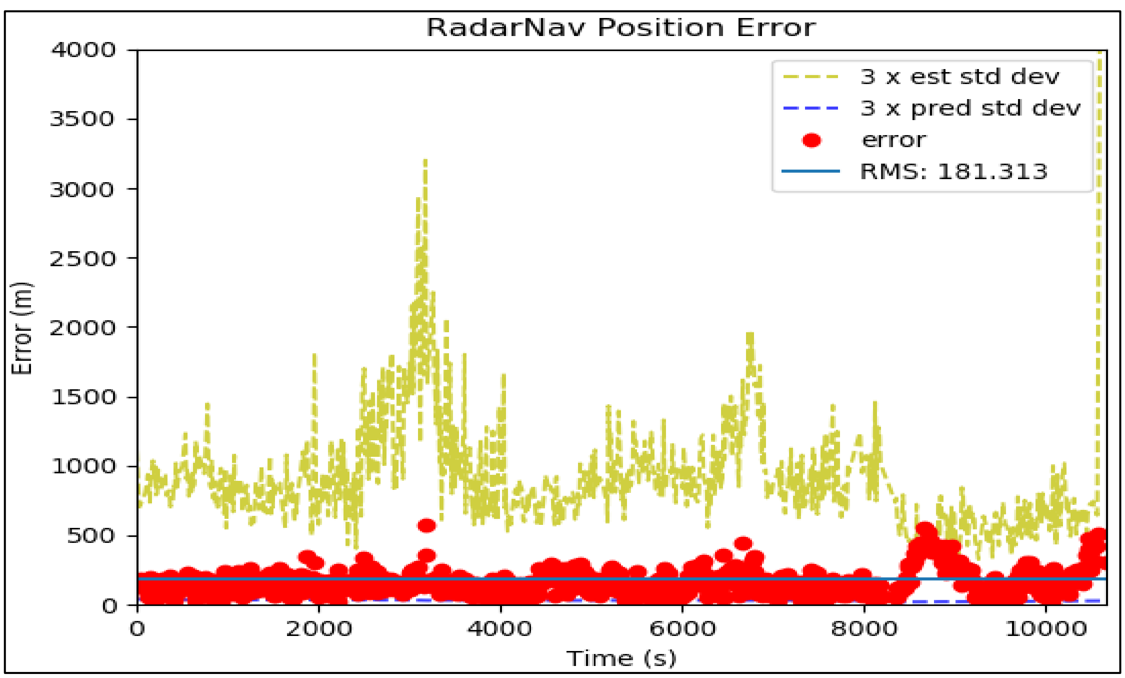

Figure 60. Plot of position errors vs. time for Kauaʻi, HI – Run 3 data set.

Figure 61 shows the same data as Figure 60 but in terms of pixels instead of distance.

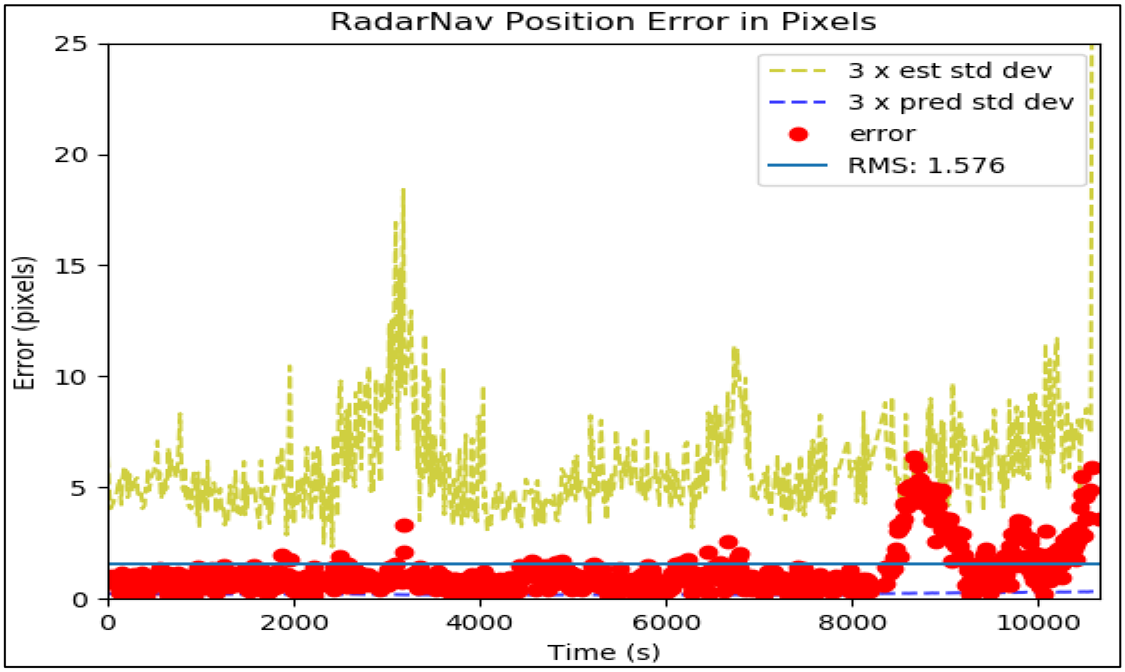

Figure 61. Plot of pixel errors vs. time for Kaua‘i, HI – Run 3 data set.

This page is intentionally blank.

6 FUTURE APPLICATIONS

Although RadarNav has been shown to provide position solutions for littoral surface vessels operating in an area where GPS is not available, there are still improvements that can and should be made prior to it being implemented in an operational environment. The next few sections cover some of these improvements, and considerations that should be taken.

6.1 MINIMUM COMPUTER SPECS FOR REAL-TIME OPERATION

RadarNav is designed to run in real time, which requires quite a bit of image processing in between each new radar sweep. RadarNav was also designed to have low overhead, so it should be able to operate on hardware that is already on board. Additionally, RadarNav has potential applications on autonomous surface vessels that may not have space for large computer racks. With all this in mind, it is useful to see how RadarNav operates on different types of hardware with a range of processer speeds and computational powers. Table 2 shows a comparison of RadarNav's average processing times on different computer platforms. The processing time is the amount of time RadarNav requires to process each new radar image. It is also the time between estimated position updates. As expected the processing times get smaller as the available processing power increases.

Table 2. Comparison of RadarNav's speed on various computer systems.

Computer	Processor	Processor Speed (GHz)	Footprint (in.^{2})	Ave. Processing Time (sec)
HP® Laptop	Intel® Core™ i7-4800MQ	2.70	150	5.5915
Intel® NUC	Intel® Core™ i7-8559U	4.50	20.3	4.1186
NVIDIA® Jetson Nano™	ARM Cortex A57	1.43	12.1	17.9313

Additionally, Table 2 shows that RadarNav can be run on a variety of platforms. The Intel® Next Unit of Computing (NUC) is basically a full PC in a miniaturized form factor. This would be great for smaller boats or autonomous surface vessels. If the operation can deal with slower update rates, then an even smaller option is available with the Jetson Nano™. This is a GPU-backed single-board computer that is a little bigger than a credit card. It was designed to bring machine learning applications to hobbyists and other applications where Size, Weight, Power and Cost (SWaP-C) is a concern. Depending on the speed of the vessel, position updates more than twice per minute could be perfectly acceptable.

6.2 NATIVE OPERATING SYSTEM INSTEAD OF ROS

The Robot Operating System (ROS) is a flexible framework of software tools and libraries designed for collaboration and functionality across a wide range of robotic platforms. ROS enables the RadarNav to be separated into multiple executables, or nodes, that are able to quickly communicate with one another through the ROS filesystem. With ROS, the nodes can be seamlessly modified and debugged without impacting others. The original decision to build RadarNav within ROS was based on the desire to move the project forward quickly and prove that the concept could work. ROS was well understood by project engineers and pieces of code that were previously written in ROS could be reused for RadarNav. Additionally, ROS is very useful for message handling, debugging and importing data from external sensors (in this case, the radar) in real time.

Although it was beneficial in the beginning, ROS has several weaknesses that are currently holding RadarNav back. The most limiting of these weaknesses is in its filesystem, which limits RadarNav ability to move large amounts of data between the nodes. This is critical when passing maps between nodes, because it limits the size of the map or map tile that can be sent for processing. Of course, this is especially apparent when the radar is in a map tile that is mostly water, but the land it is “seeing” is in a different tile.

A less critical issue with ROS would come when introducing it to an operational environment where Information Technology (IT) security is a big concern. ROS represents one more application that must be updated, monitored and patched throughout its lifecycle. Running the RadarNav application as its own entity places ownership entirely in the hands of the Navy which would increase security and reduce lifecycle maintenance.

6.3 MAP RELATED IMPROVEMENTS

Although RadarNav has always relied on pretty much the same map sources, there are still some tweaks that needed to be made manually, and the map quality is not consistent across all maps (specifically CONUS vs OCONUS). These maps all come from publicly available sources. However, in an operational environment the map sources would be different and possibly classified, so the map delivery system would need to be updated. It would be beneficial to automate this process and allow for the use of maps from multiple sources that may be operationally relevant; especially in terms of the classification levels.

A useful idea that was never implemented into RadarNav was to have it host its own map server. This would allow seamless flow from one map tile to the next. It should also limit RadarNav’s search area to match the range of the radar rather than the area of a map tile. This would reduce cold-start times when no initial position is available and allow the search area to cover multiple tiles without overloading the available memory with map information outside of what the radar can “see”. The map server should also allow for varying map resolutions. Instead of the current method of limiting by pixel dimensions, the search area should be limited by nautical miles or kilometers, regardless of the resolution. This would be especially helpful when it comes to higher resolution maps. Of course, in order to take advantage of the higher resolution maps, the radar images would also need to be higher resolution which would increase RadarNav’s processing times. A map server would also be necessary for an operational setting. The transits presented in this report are kept to a single map tile; but for some datasets multiple map tiles were manually merged and/or cropped to ensure that the entire transit fit within a single tile and the search area was kept to a reasonable size. Using a map server would prevent the need to manually manipulate map tiles, allow for longer transits and allow RadarNav to operate without the need to know which map tile to load ahead of time.

6.4 NEED ACCESS TO VESSEL’S RADAR SYSTEM

When implementing RadarNav in any operational environment, it will need access to the vessel’s radar system. The data collections and real-time demonstrations presented in this report show that RadarNav can be very flexible when it comes to receiving radar data. This is very important because there are so many different types of radar systems and so many variations for their implementation. In other words, connecting to a vessel’s radar system is highly platform specific. The good news is that modern radar systems are digital, so the worst-case scenario would be that a driver would need to be written to translate data between each radar system and RadarNav. Of course, this would be for cases where RadarNav was being permanently installed and/or it was desirable to provide RadarNav the digital radar data directly from the radar. This is the most robust solution, and is the method that was used for the real-time demonstrations along the San Diego coast presented in this report. The collection method used for the Charleston data is also a good choice for a permanent installation. In this case, the radar data was transmitted via the vessel's network. The more universal approach that was used for the rest of the data presented in this report was to grab snapshots from the video feed to the radar's display. This requires additional processing in order to crop the snapshot, which would vary from one display to another. Although this could possibly be done in real time, special care would need to be given to cases where the display is changed or not showing the standard radar screen. In these cases, RadarNav may not function.

6.5 TIE IN WITH OTHER NAVIGATION SYSTEMS

There is currently no such thing as a single foolproof navigation system. Global Navigation Satellite Systems (GNSS) such as the Global Positioning System (GPS) have been the gold standard for quite a while, but a GNSS is not always available either due to environmental or adversarial reasons. Modern navigation systems are leaning towards multiple sensors and systems being fused together to provide the user with the best possible position solution. One such system currently under development is the Global Positioning System-Based Positioning, Navigation, and Timing Services (GPNTS), which aims to provide 3D position, velocity, acceleration, attitude and timing information, gathered from multiple sensors and sources. All of these systems have their limitations (even GPS), so it is crucial that they work together to deliver accurate positions. RadarNav is no different. It has its own limitations and cannot perform in every situation. Therefore, it would be advantageous to include RadarNav in multi-sensor navigation systems on platforms that already have radar systems installed. The benefit of RadarNav is obviously having another navigation system to provide more accurate position solutions. The cost of RadarNav is fairly low since it takes advantage of radar systems already installed and in use. RadarNav also has fairly low maintenance costs because it is a software-based system and the only hardware unique to RadarNav is the computer it runs on.

Another advantage of RadarNav is that the output can be configured to comply with the interface required of GPNTS or any other digital system. In addition to the position solution, RadarNav also provides a confidence metric that gives users an idea of how accurate each position solution is in any given scenario. This provides the user with improved decision-making capabilities, and would be especially useful for autonomous surface vessels.

This page is intentionally blank.

7 CONCLUSIONS

7.1 LIMITATIONS OF RADARNAV

The obvious major limitation of RadarNav is that the radar must be able to "see" the coast or enough other navigable features to provide reasonable position estimates. This is true of any type of map matching navigation technique. More powerful radar systems will be able to operate further away from the shore, but at some point, the curvature of the earth prevents the radar from "seeing" anything but open ocean.

As shown in the data collected from Sweden, being too close to the coast can also cause problems for RadarNav if the geographic features along the coast are so close and tall that they block the radar from "seeing" anything else. Fortunately, this situation usually only occurs for a limited time and RadarNav recovers fairly quickly. Coupling RadarNav with an Inertial Navigation System (INS) for dead reckoning, would eliminate this issue.

RadarNav's other limitations are largely hardware or input dependent and can be viewed as a trade-space in operational settings. Map quality certainly impacts RadarNav's accuracy, but Naval vessels typically have access to very good maps. The quality of the radar, in terms of power, resolution, signal-to-noise, etc. is equally as important as the quality of the maps. This hardware limitation may be addressed by upgrading the radar system. However, it may be more practical to simply accept this limitation.

Fortunately, RadarNav's availability analysis tool can be used to foresee problems that may arise from these limitations prior to the operation. Alternatively, RadarNav's real-time estimated error provides a confidence metric while under way so warfighters or autonomous systems have a reasonable expectation about how much to trust RadarNav's position estimate at any given time.

7.2 TEST DATA SUMMARY AND COMPARISON

The strength, resolution and quality of the radar system can have a dramatic effect on RadarNav's accuracy. Since RadarNav works by matching real radar images to radar images synthesized from elevation maps, any variation between the images will increase the error of the position solution. Although RadarNav is able to model certain aspects of the radar in use (e.g. height, power, resolution), sometimes the real radar images just don't have enough information to allow for proper matching. Table 3 summarizes the results from the data collections and real-time demonstrations described in Section 1. This is a convenient side-by-side comparison of RadarNav's results. As expected, the actual error (versus GPS position) falls between the Availability Analyzer's predicted error and the real-time estimated error. On average, the estimated error was about three times higher than the actual error, and the predicted error was about 18 times lower than the actual error. This means that the scaling of the predicted error could be changed to provide errors that are closer to real world values. However, a better solution would be to improve the modeling of the radar system. This would also improve the actual error as well. Unfortunately, these results are not directly comparable because the radar systems used at each location were different and the distance to land masses was quite varied. The data from Norfolk and that taken from the Kilo Moana have the worst errors, but this is due to the fact that the radar was farther away from shore than in the other data sets. In Norfolk, the radar was about 2Km away from shore, and the Kilo Moana was rarely closer than 5Km from shore. In the other data sets, the radar was typically less than 0.5Km from shore. In contrast, the Sweden data sets show the best results because they were typically very close to shore, and the radar could "see" land all around it, rather than just on one side.

Table 3. Side-by-side comparison of position error test results.

Location/Event	Availability Analyzer Predicted RMS Position Error (m)	Actual RMS Position Error (m)	Covariance Estimated RMS Position Error (m)
Red Beach, MCCS Camp Pendleton, CA	4.852	92.825	249.559
San Diego, CA	10.632	74.151	214.965
Norfolk, VA	113.053	961.258	1045.064
Sweden – Run 1	1.476	35.469	107.539
Sweden – Run 2	1.739	57.753	112.152
Sweden – Run 3	2.952	62.83	129.347
Sweden – Run 4	2.437	47.133	100.754
Astoria, OR - Columbia River Delta	55.176	389.372	436.095
Honolulu, HI – Run 1	29.659	481.437	1733.247
Honolulu, HI – Run 2	18.855	255.828	1231.743
Kaua‘i, HI – Run 1	29.092	204.832	1473.175
Kaua‘i, HI – Run 2	46.089	261.302	1138.839
Kaua‘i, HI – Run 3	30.971	181.313	1069.86

Figure 62 and Figure 63 illustrate why the quality of the radar system affects the position error. Figure 62 shows a typical radar image for the San Diego area using a Furuno FAR-2117 radar with a 6.5 ft antenna (left image), compared to a silhouette of the land in the same area (right image). It should be noted that the silhouette is not the synthesized image that RadarNav uses for comparison. It is being used only for illustrative purposes in this report. The synthesized image would show occlusions and look much more like the radar image on the left. In fact, the occlusions are responsible for providing information on the position. In the radar image in Figure 62 (left image), the occlusions show that the radar was somewhere in the bottom-left hand corner of the image because the land that is causing the occlusions is between that corner and the missing or occluded areas of the image.

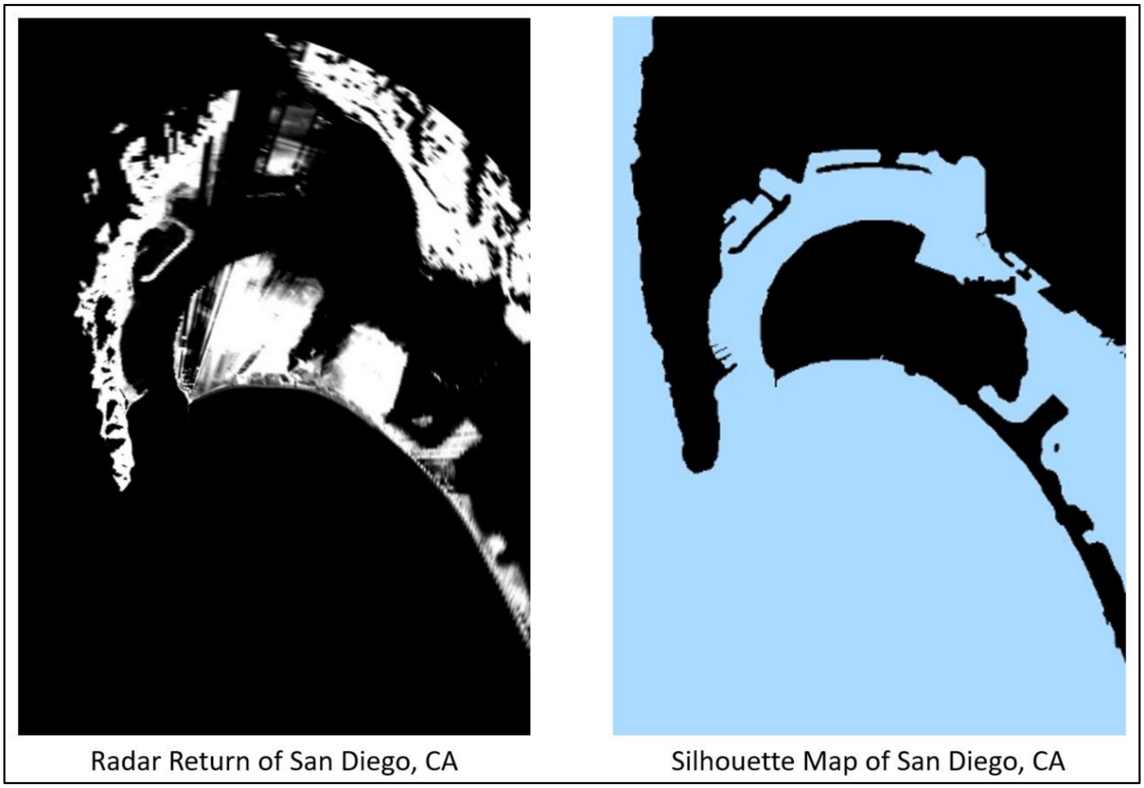

Figure 62. Comparison of San Diego coastal outline to 12Kw radar return.

Both images in Figure 62 look visually similar and it is easy to tell that they represent the same land mass. However, in Figure 63, the real radar image (left) and the silhouette of the land (right) are not visually similar at all. The images in Figure 63 represents Norfolk and Hampton, VA. The poor quality of the radar image (left) is due to the lower power of the radar being used and the lack of tall geographical features in the area. Once occlusions are added based on the elevation maps and radar modeling parameters there is not much information left in the images. This makes it difficult for RadarNav to differentiate between radar images from neighboring positions, and drives the RMS error up.

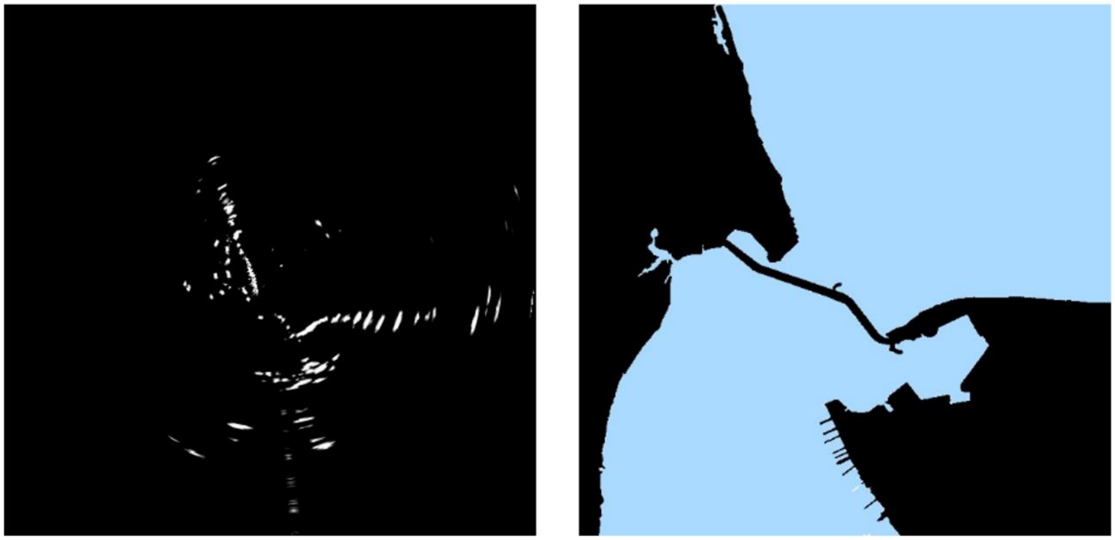

Figure 63. Comparison of Norfolk coastal outline to 4Kw radar return.

Table 4 provides a side-by-side comparison of the radar systems used at each location mentioned in this report. This table lists the maximum power and range possible for each system, but these are configurable, so it doesn't mean that the radar was set to maximum power and maximum range while collecting the data presented in this report. A more powerful radar with longer range is capable of seeing land from farther distances, which would defiantly allow RadarNav to provide more accurate position estimates. However, other factors such as noise and resolution also impact the position estimates.

Table 4. Side-by-side comparison of radar equipment used.

Location	Radar Brand & Model	Radar Frequency Band	Radar Power – Max (Kw)	Radar Antenna Length (ft)	Radar Range – Max (nm)
Red Beach, MCCS Camp Pendleton, CA San Diego, CA	Furuno FAR-2117	X-Band	12	6.5	96
Norfolk, VA	Furuno DRS4A	X-Band	4	3.5	48
Sweden	Furuno DRS6A	X-Band	6	4	120
Astoria, OR Honolulu, HI Kaua'i, HI	Furuno FAR-3230	S-Band	30	12	96

Since RadarNav depends on image processing and relating image distances to real-world distances, comparing results from datasets with different map or radar image resolutions can be misleading. If all other factors were equal but the resolution of Data set A had more meters per pixel (m/p) than Data set B, the distance of position errors in Data set A would be higher. In the real world this wouldn't matter because the data sets are not being compared and the only thing that matters is how far off RadarNav's position estimate is from the real-world position. However, this comparison is useful for research purposes, and it is important to understand why the error would appear larger even if the hardware, location and algorithm didn't change. Table 5 shows a comparison of the position error and pixel error for each location. Table 5 also lists the meters per pixel resolution of the map used at each location, and the maximum power of the radar used at each location.

Table 5. Side-by-side comparison of position errors and pixel errors.

Location	Resolution (m/p) †	Radar Range (Km) †	RMS Position Error (m)	RMS Pixel Error (pixels)‡
Red Beach, MCCS Camp Pendleton, CA	38.6	9.9	92.8	3
San Diego, CA	38.6	9.9	74.2	2
Norfolk, VA	60.47	14.8	961.3	16
Sweden*	16.3	4.1	50.8	4
Astoria, OR	188	44.4	389.4	3
Honolulu, HI*	130.2	33.3	368.6	4
Kaua'i, HI*	173.6	44.4	215.8	2

*Values represent an average for each of the runs at the location indicated

†Values represent maximum value used during run

‡RMS pixel errors rounded up

If the RMS pixel error was constant between datasets with different resolutions, then the limiting factor would be the resolution and/or something internal to the RadarNav algorithm causing a pixel uncertainty. However, since the pixel error changes between datasets of different resolutions the quality of the radar (which is manifested as the amount of land the radar "sees") must also be contributing to the error. However, a more powerful radar will not result in smaller errors if there isn't enough land to make comparisons, or the land is just too far away.

7.3 APPROPRIATE ENVIRONMENTS FOR RADARNAV

RadarNav is intended for surface vessels equipped with navigation radar, operating in littoral areas. It can be run on computers of varying processing power depending on the update rate required for the specific operation. RadarNav relies on existing radar systems already installed on the surface vessel, so no additional hardware is required to implement RadarNav unless the decision is made to run RadarNav on a its own dedicated computer. Even if this is the case, the cost of implementation and maintenance is very low. RadarNav can be used in either autonomous or man-in-the-loop situations. RadarNav is capable of providing a position solution in most littoral and riverine areas and would be a benefit to multi-sensor navigation systems operating in regions where GPS may not be reliable or available.

This page is intentionally blank.

REFERENCES

[1] D. S. P. S. B. Jignesh N Sarvaiya, "Image registration using Log polar transform and phase correlation," *TENCON 2009 - 2009 IEEE Region 10 Conference*, pp. 1-5, 2009.

[2] U. S. G. S. (USGS), "The National Map Viewer," United States Geological Survey (USGS), [Online]. Available. https://viewer.nationalmap.gov/advanced-viewer/. [Accessed 11 September 2020].

[3] N. G.-I. A. (NGA), "Digital Nautical Charts (DNC)," National Geospatial-Intelligence Agency (NGA), [Online]. Available. https://dnc.nga.mil/. [Accessed 11 September 2020].

[4] U. S. G. S. (USGS), "EarthExplorer," United States Geological Survey (USGS), [Online]. Available. https://earthexplorer.usgs.gov/. [Accessed 11 September 2020].

[5] N. C. F. E. I. (NCEI), "GSHHG. A Global Self-consistent, Hierarchical, High-resolution Geography Database," National Centers For Environmental Information (NCEI), [Online]. Available. http://www.soest.hawaii.edu/pwessel/gshhg/. [Accessed 11 September 2020].

[6] Wikipedia, "RV Kilo Moana (T-AGOR-26)," Wikipedia, [Online]. Available. https://en.wikipedia.org/wiki/RV_Kilo_Moana_(T-AGOR-26). [Accessed 11 September 2020].

This page is intentionally blank.

INITIAL DISTRIBUTION

84310	Technical Library/Archives	(1)
71780	M. Bozeman	(1)
71780	M. Nguyen	(1)
71740	M. Alam	(1)

Defense Technical Information Center Fort Belvoir, VA 22060–6218	(1)
Office of Naval Research (ONR)	(1)

This page is intentionally blank.

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-01-0188*

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
October 2021	Final	

4. TITLE AND SUBTITLE

Radar Navigation Performance and Final Report

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

~~**5d. PROJECT NUMBER**~~

5e. TASK NUMBER

5f. WORK UNIT NUMBER

6. AUTHORS

Eric Bozeman
Minhdao Nguyen
Mohammad Alam
NIWC Pacific

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

NIWC Pacific
53560 Hull Street
San Diego, CA 92152–5001

8. PERFORMING ORGANIZATION REPORT NUMBER

TR 3248

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

Office of Naval Research (ONR)
875 N. Randolph St, Arlington, VA 22217

10. SPONSOR/MONITOR'S ACRONYM(S)

ONR, Code 312

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

12. DISTRIBUTION/AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
Approved for public release. Distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report compiles the work done throughout the course of the Radar Navigation (RadarNav) project. RadarNav was funded by Dr. Tommy Willis of ONR Code 312, Precision Navigation and Timekeeping. The work was carried out by Naval Information Warfare Center (NIWC) Pacific; formally Space and Naval Warfare Systems Center (SPAWAR) Pacific. RadarNav is a navigation solution for surface vessels in littoral areas without access to GNSS (Global Navigation Satellite Systems). This report includes a detailed description of the RadarNav system as well as the results of several demonstrations and data collection events. Additionally, there are recommendations for the future use and implementation of RadarNav, which includes possible improvements to the system.

.

15. SUBJECT TERMS

RadarNav; GNSS; PNT; LIDAR navigation systems; Robot Operating System; Digital Elevation Models; radar

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Eric Bozeman
					19B. TELEPHONE NUMBER *(Include area code)*
U	U	U	SAR	88	(619) 553-1044

Standard Form 298 (Rev.
Prescribed by ANSI Std. Z39.18

This page is intentionally blank.

This page is intentionally blank.

Naval Information Warfare Center Pacific (NIWC Pacific)
San Diego, CA 92152-5001