Satellite Products and Services Review Board

Algorithm Theoretical Basis Document:

IWRAP Near-Real-Time Visualization System

Compiled by:
NOAA/NESDIS/STAR Ocean Surface Winds Team

Version 1.0

April, 2025

AUTHORS:

Paul S. Chang (NOAA/NESDIS/STAR) paul.s.chang@noaa.gov
Zorana Jelenak (UCAR) zorana.jelenak@noaa.gov
Casey Shoup (Global Science & Technology) casey.shoup@noaa.gov
Joe Sapp (Global Science & Technology) joe.sap@noaa.gov

DOCUMENT HISTORY DOCUMENT REVISION LOG

DOCUMENT TITLE: ATBD – IWRAP NRT Visualization System			
DOCUMENT CHANGE HISTORY			
Revision No.	**Date**	**Revision Originator Project Group**	**CCR Approval # and Date**
1.0	04/2025	NOAA/NESDIS/STAR OSWT	N/A

LIST OF CHANGES

DOCUMENT TITLE: ATBD – IWRAP NRT Visualization System					
LIST OF CHANGE-AFFECTED PAGES/SECTIONS/APPENDICES					
Version Number	**Date**	**Changed By**	**Page**	**Section**	**Description of Change(s)**
1.0	04/2025	Z.Jelenak	All	All	Initial document

TABLE OF CONTENTS

LIST OF ACRONYMS

AOC: Aircraft Operations Center

ATBD: Algorithm Theoretical Basis Document

AVAPS: Airborne Vertical Atmospheric Profiling System

dBZ: Decibels of Reflectivity Factor

IWRAP: Imaging Wind and Rain Airborne Profiler

IWG: In-flight Weather Data (IWG-1 format)

KaIA: Ka-band Interferometric Altimeter

NESDIS: National Environmental Satellite, Data, and Information Service

NHC: National Hurricane Center

NOAA: National Oceanic and Atmospheric Administration

NRT: Near-Real-Time

OSVW: Ocean Surface Vector Winds

SFMR: Stepped Frequency Microwave Radiometer

SOCD: Satellite Oceanography and Climatology Division

STAR: Center for Satellite Applications and Research

SWH: Significant Wave Height

VAD: Velocity-Azimuth Display

VDM: Vortex Data Message

WGS84: World Geodetic System 1984

Ze: Equivalent Radar Reflectivity Factor

Chapter 1

IWRAP Instrument and System Overview

I. Introduction

This Algorithm Theoretical Basis Document (ATBD) describes the algorithms and processing methodology used in the Imaging Wind and Rain Airborne Profiler (IWRAP) Near-Real-Time (NRT) Visualization System. The system generates a suite of static and interactive visualizations from airborne Doppler radar and ancillary meteorological data collected during NOAA hurricane reconnaissance flights. These products are disseminated through the NOAA/NESDIS/STAR Aircraft Reconnaissance Data web portal.

The IWRAP is a dual-beam, dual-wavelength (Ku-band) conically scanning airborne Doppler radar that measures three-dimensional wind profiles beneath the aircraft. It operates aboard the NOAA P-3 hurricane hunter aircraft and provides vertically resolved wind speed, wind direction, vertical wind velocity, and radar reflectivity (Ze) from flight level to the ocean surface. The radar uses a Velocity-Azimuth Display (VAD) technique to retrieve horizontal wind vectors at multiple height levels from spectral analysis of the Doppler returns.

II. System Architecture

The IWRAP NRT Visualization System is a Python-based processing pipeline managed through a Snakemake workflow engine and a Conda environment (iwrap-nrt-plots) that ensures reproducibility. The system is organized into standalone visualization modules, each implemented as a command-line Python script using the Click library. A shell orchestrator (run_figs.sh) coordinates execution of multiple modules for a given flight pass, accepting parameters for flight ID, pass number, radar band (ku1, ku2, ku4, or kub for blended), path direction (inbound/outbound), pass count, and storm name.

The end-to-end processing flow consists of the following stages: (1) Data Ingestion of IWRAP NetCDF files, IWG-1 flight metadata, VDM storm center reports, dropsonde profiles, best-track data, and land mask datasets; (2) Variable Normalization to standardize naming across processing versions; (3) Quality Filtering to remove invalid measurements; (4) Coordinate Transformation using WGS84 geodesic calculations; (5) Unit Conversion from SI to operational meteorological units; (6) Spatial Interpolation for interactive products; (7) Visualization Rendering using Matplotlib, Cartopy, Plotly, and Bokeh; (8) Thumbnail Generation via Pillow; and (9) Dissemination via SCP to the web server.

III. Input Data

The primary input is IWRAP radar data in NetCDF format. Each file corresponds to a single flight pass and radar band combination with naming convention: {flight_id}_{band}_pass{pass_num}_combined.nc. Table 1.1 lists the key variables contained in these files.

Table 1.1: IWRAP NetCDF Input Variables

Variable	**Units**	**Description**
spec_hws	m/s	Spectral horizontal wind speed
spec_hwdir	degrees	Spectral horizontal wind direction (from north)
spec_w	m/s	Spectral vertical wind velocity
Ze	dBZ	Equivalent radar reflectivity factor
spec_hws_std	m/s	Horizontal wind speed standard deviation
sfmr_wind_speed	m/s	Collocated SFMR surface wind speed
sfmr_rain_rate	mm/hr	SFMR-derived rain rate
fl_wind_speed	m/s	Flight-level wind speed
fl_wind_dir	degrees	Flight-level wind direction
radar_altitude	m	Aircraft altitude above surface
lat, lon	degrees	Geographic coordinates
height	index	Height above surface index
az_gap	degrees	Azimuthal gap in radar coverage
resid_rms	—	RMS residual of spectral retrieval

Additional data sources include: IWG-1 flight metadata text files providing aircraft navigation and meteorological parameters; Vortex Data Messages (VDM) providing storm center fix positions; AVAPS dropsonde profiles in ASCII format providing independent vertical wind measurements; best-track forecast data for storm context; a high-resolution land mask (landmask.nc) for ocean/land filtering; OSVW scatterometer NetCDF files for C-band and Ku-band surface wind retrievals; and KaIA significant wave height retrievals in CSV format.

IV. Software Dependencies

The system requires Python >= 3.10 and the following core packages: NumPy, xarray, NetCDF4, Pandas, Matplotlib, Cartopy, Basemap, Plotly, SciPy, GeographicLib, Click, Pillow, Snakemake, cmocean, Pint, and Seaborn. Three in-house Git submodules are required: oswtcolor (NOAA OSWT color mapping utilities), util (spherical geodesy and angle calculations), and aircraft (IWG-1 data readers).

Chapter 2

IWRAP Wind Profile Visualization Algorithm

I. Introduction

The IWRAP wind profile visualization modules generate time-height cross-sections and three-dimensional representations of the wind and reflectivity structure observed during hurricane reconnaissance passes. This chapter describes the core algorithms applied to transform raw IWRAP data into calibrated, georeferenced visualizations.

II. Variable Normalization

The iwrap_var_check() function standardizes variable naming across different versions of the IWRAP processing chain. This includes mapping alternative names (e.g., hws to spec_hws, hwdir to spec_hwdir, w to spec_w) and handling variant quality metric names (azp_gap to az_gap, rms_res to resid_rms). This normalization ensures backward compatibility as the upstream processing evolves.

III. Height Coordinate Conversion

IWRAP height coordinates are stored as dimensionless indices. The conversion to physical height units follows:

$$height\_km = (height\_index\ /\ 30) \times 1000$$

This yields height above the ocean surface in meters. The scaling factor of 30 corresponds to the number of range bins per kilometer in the IWRAP radar sampling geometry.

IV. Unit Conversions

All wind speed values are converted from meters per second to nautical knots for consistency with operational conventions:

$$V\_knots = V\_m/s \times 1.94384$$

KaIA significant wave height is converted from meters to feet: $SWH\_ft = SWH\_m \times 3.28084$.

V. Wind Vector Decomposition

Horizontal wind speed and direction are decomposed into U (east-west) and V (north-south) components using standard meteorological convention, where wind direction is the direction from which the wind blows:

$$U = -V\_speed \times \sin(-\theta)$$

$$V = V\_speed \times \cos(-\theta)$$

where θ is the wind direction in radians measured clockwise from north. The wind direction in degrees is recovered via $\theta\_dir = \arctan2(U, V)$.

VI. Geodesic Storm-Relative Coordinate Transformation

Storm-relative coordinates are computed using the WGS84 ellipsoid via GeographicLib. For each measurement, the geodesic inverse problem is solved to obtain the distance (d) and forward azimuth (α) from the observation location to the interpolated storm center:

$$(d, \alpha) = Geodesic.WGS84.Inverse(lat\_obs,\ lon\_obs,\ lat\_center,\ lon\_center)$$

VII. Radial and Tangential Wind Decomposition

Wind vectors are decomposed into radial (storm-inflow/outflow) and tangential (rotational) components relative to the storm center using the azimuth angle α:

$$V\_tangential = -U \times \sin(450^\circ - \alpha) + V \times \cos(450^\circ - \alpha)$$
$$V\_radial = U \times \cos(450^\circ - \alpha) + V \times \sin(450^\circ - \alpha)$$

Positive tangential wind indicates cyclonic flow (counterclockwise in the Northern Hemisphere), while positive radial wind indicates outflow away from the storm center.

VIII. Storm Center Temporal Interpolation

VDM-reported storm center positions are available at discrete, typically hourly, intervals. To obtain the storm center position at each IWRAP measurement time, the VDM fix positions are interpolated linearly in time. The VDM module ingests storm center latitude, longitude, and fix timestamps, and a one-dimensional linear interpolation (scipy.interpolate.interp1d) is applied independently to the latitude and longitude coordinates as functions of time.

IX. Quality Filtering

Multiple quality control filters are applied to the IWRAP data prior to visualization:

- Zero Wind Speed Filter: Measurements with zero horizontal wind speed are excluded as they typically indicate retrieval failure.
- Standard Deviation Filter: Measurements without a valid wind speed standard deviation (spec_hws_std) are excluded.
- Land Mask Filter: Measurements over land surfaces are identified using the land mask dataset and Basemap is_land() queries, then excluded from ocean-only products.

- Coordinate Validity Filter: Measurements with non-finite latitude or longitude values, or with anomalous position jumps (greater than 0.1 degrees between consecutive points), are removed.
- SFMR Heavy Rain Filter: SFMR measurements during heavy rain events (rain_rate > 8 mm/hr) are flagged on profile visualizations.

X. Spatial Interpolation

The interactive visualization modules employ SciPy griddata with linear interpolation to map irregularly sampled IWRAP observations onto a regular two-dimensional grid in time-height space. This fills gaps in radar coverage while preserving the structure of the observed wind and reflectivity fields.

XI. Colormap Specifications

Custom piecewise-linear colormaps are defined for radar reflectivity and wind speed. The Ze colormap spans 0–75 dBZ with color transitions: gray (0–5), cyan-to-blue (5–20), green-to-dark green (20–35), yellow-to-gold (35–45), orange-to-red (45–60), dark red-to-magenta (60–70), and white (70–75). Boundary normalization at 1-dBZ intervals is applied; interactive products use 10× finer color bins.

The wind speed colormap spans 0–160 knots aligned with the Saffir-Simpson Hurricane Wind Scale: light gray (0–7 kt), cyan-to-teal (7–25), teal-to-dark teal (25–34), green-to-yellow (34–64), orange-to-red (64–96), and purple-to-pink (96–160). The operational NOAA OSWT colormap (oswtcolor) is used for storm summary products with ticks at 0, 7, 16, 25, 34, 40, 46, 52, 58, 64, 80, 96, 110, 125, 140, 155, and 160 knots.

XII. Output Products

The profile visualization module (iwrap_fig_profile.py) generates interactive Bokeh HTML files with time-height cross-sections of wind speed, direction, vertical wind, and Ze. The five-panel static module (iwrap_fig_5p-static.py) generates multi-panel PNG images with synchronized time axes. The five-panel interactive module (iwrap_fig_5p-inter.py) enhances these with spatial interpolation and dropsonde overlays. The 3D modules (iwrap_fig_3d-static.py and iwrap_fig_3d-inter.py) create three-dimensional scatter plots in longitude-latitude-altitude space, color-coded by wind speed or reflectivity.

XIII. Limitations

- Linear temporal interpolation of storm center position assumes quasi-steady storm motion between VDM fix times.

- Height coordinate conversion assumes a fixed range-bin spacing; changes in IWRAP sampling configuration require scaling factor updates.
- The land mask assumes a static coastline and does not account for storm surge inundation.
- Spatial interpolation in interactive products uses linear interpolation, which may smooth sharp gradients such as eyewall wind maxima.

XIV. References

[1] Fernandez, D. E., et al., 2005: IWRAP: The Imaging Wind and Rain Airborne Profiler. *IEEE Trans. Geosci. Remote Sens.*, 43(8), 1775–1787.

[2] Uhlhorn, E. W., and P. G. Black, 2003: Verification of remotely sensed sea surface winds in hurricanes. *J. Atmos. Oceanic Technol.*, 20(1), 99–116.

[3] NOAA Aircraft Operations Center, IWG-1 format specification.

[4] Reasor, P. D., et al., 2009: Vortex Data Message (VDM) specification, National Hurricane Center.

Chapter 3

OSVW Wind Retrieval Visualization Algorithm

I. Introduction

The Ocean Surface Vector Wind (OSVW) visualization module (osvw_plots.py) produces comparison plots of C-band and Ku-band surface wind retrievals from the airborne scatterometer alongside reference measurements from SFMR and flight-level instruments. This chapter describes the ambiguity resolution algorithm and associated processing.

II. OSVW Ambiguity Resolution Algorithm

The OSVW scatterometer retrieval produces two ambiguous wind direction solutions for each measurement. The ambiguity resolution algorithm selects the solution closest to the flight-level wind direction reference:

(1) For each measurement, the circular wind direction difference between each ambiguity solution and the flight-level direction is computed. The circular difference accounts for the 360-degree wrap-around: diff = 180 – |180 – |d1 – d2||.

(2) The solution with the smallest direction difference is selected, provided the difference is less than 100 degrees.

(3) A 180-degree correction is applied to convert from scatterometer convention to meteorological convention: direction_met = (direction_scat + 180) mod 360.

III. Rain Contamination Filtering

For C-band OSVW retrievals, an additional rain contamination filter is applied: measurements where rain_rate exceeds 4 mm/hr are set to NaN to avoid rain-corrupted wind estimates. Ku-band retrievals are less sensitive to rain contamination and do not require this filter.

IV. Algorithm Implementation

The module reads C-band and Ku-band NetCDF files (*uc_outer_lfm*combined.nc and *ku_outer_lfm*combined.nc, respectively), extracts both ambiguity solutions (OSVW_wind_speed_cmod5.h, OSVW_wind_dir_cmod5.h), applies the ambiguity resolution logic using flight-level wind direction as truth, filters for rain contamination, and generates time-series comparison plots and geographic map panels with wind barbs at measurement locations.

V. Output Products

Output includes time-series panels comparing C-band and Ku-band OSVW with SFMR and flight-level reference winds, map views with wind barbs at measurement locations, and ambiguity-resolved versus individual ambiguity solutions. Dropsonde locations are overlaid when available.

VI. Limitations

- Ambiguity resolution uses flight-level wind direction as reference truth, which may differ from true surface wind direction in regions of strong directional shear.
- The 100-degree threshold was empirically determined and may require adjustment for different storm environments.
- Rain-rate threshold of 4 mm/hr for C-band filtering is empirically determined.

VII. References

[1] NOAA/NESDIS/STAR Ocean Surface Winds Team internal documentation.

Chapter 4

Storm Summary and Composite Products

I. Introduction

The storm summary module (storm_summary.py) generates composite maps combining data from multiple reconnaissance flights for a single tropical cyclone. The flight path module (iwrap_fig_path.py) generates geographic visualizations of individual flight pass patterns. The thumbnail generator (iwrap_thumbnails.py) creates preview images for the web portal.

II. Storm Summary Algorithm

The storm summary module iterates over all available flight IDs for a given storm and season, reading IWG-1 data for each flight to obtain aircraft coordinates and collocated measurements. Three product variants are generated:

SFMR Surface Wind Summary: SFMR wind speed is converted from m/s to knots. Values below 30 knots are masked to highlight tropical storm and hurricane-force winds. Each flight track is scatter-plotted on a Cartopy PlateCarree projection with wind speed as the color variable.

Flight-Level Wind Summary: Flight-level wind speed (WS.d) is converted to knots and displayed similarly.

KaIA Significant Wave Height Summary: KaIA CSV files are concatenated into a Pandas DataFrame, SWH is converted from meters to feet, and the data are scatter-plotted along the flight track using the oswtcolor altimeter colormap.

Flight track lines with black-bordered white fill provide spatial context, and each flight timestamp is annotated at the track midpoint.

III. Flight Path Visualization

The flight path module reads pickle files containing pass location dictionaries, then plots each reconnaissance pass in a distinct color from the Dark2 qualitative colormap on a Cartopy PlateCarree map with 10m-resolution land and lake features. Pass labels are shown in a sorted legend.

IV. Thumbnail Generation

The thumbnail generator uses Pillow to create 500×500 pixel preview versions of all output PNG images. Thumbnails are saved to a parallel thumbs/ directory structure and transferred to the web server alongside the full-resolution images.

V. Limitations

- Storm summaries assume all flights for a storm are stored under a consistent directory naming scheme.
- KaIA SWH data availability varies by flight; missing flights are skipped silently.

Chapter 5

Data Dissemination and Web Integration

I. Dissemination Architecture

All generated figures (PNG and HTML) are saved to the NOAA data archive at /data/oswt-project-archive/. After generation, files are transferred via SCP to the web-facing server under the /sftp-chroot/data_iwrap/testing/ directory hierarchy, organized by year, storm name, and flight ID. File permissions are set to 775 for group read access. Thumbnail images follow the same transfer process to a parallel thumbs/ directory.

II. Web Portal

Products are displayed on the Aircraft Reconnaissance Data web portal at https://manati.star.nesdis.noaa.gov/datasets/AircraftDataNew.php. The portal organizes products by hurricane season, storm name, and flight mission, allowing users to browse full-resolution static images, interact with HTML-based visualizations, and view storm-level summaries.

III. Workflow Automation

The processing pipeline is managed via Snakemake, which handles dependency resolution and environment reproducibility. The Conda environment specification (workflow/requirements.yaml) defines all package dependencies. The shell orchestrator (run_figs.sh) provides a single entry point for generating all visualization products for a given flight pass.

IV. Future Enhancements

- Implementation of automated quality control using machine learning classifiers.
- Integration of additional radar bands and multi-frequency retrieval techniques.
- Development of real-time data streaming capabilities to reduce latency.
- Enhanced storm-relative coordinate analysis with vortex-centered composites.
- Integration with NOAA operational forecast models for observation-forecast comparisons.