A Python Tool for Apogee IRT Sensor Footprint Generation and Georeferencing – Apogee_IRT_Footprint

Rui Gao[1,2], Mohammad Safeeq[1,2], Joshua H. Viers[1,2]

[1] Department of Civil and Environmental Engineering, University of California, Merced, CA 95343, USA
[2] Valley Institute for Sustainable, Technology & Agriculture, University of California Merced, Merced, CA 95343, USA

Abstract: This repository provides a Python-based toolkit for modeling and visualizing the ground footprint of downward- or angled-facing Infrared RadiomeTer (IRT) in geospatial coordinates. The tool computes the elliptical (or circular) footprint projected by an IRT sensor based on installation height, field-of-view (FOV), and view angle. It supports rotation to align the ellipse with the direction of observation (e.g., wind direction or sensor azimuth), and georeferences the output using a user-defined WGS84 coordinate. The final footprint can be exported as a GeoTIFF raster with user-defined resolution, where pixels inside the ellipse are assigned a value of 1, and those outside are 0. This is useful for spatial modeling, footprint-based data analysis, and remote sensing validation studies. All computations are done in the Cartesian coordinate system, projected into UTM, and visualized using standard Python libraries. This tool facilitates reproducible and accurate IRT footprint mapping across field studies, and can be extended for use in micrometeorological applications, surface energy balance models, or UAV-based thermal remote sensing in both agricultural and ecosystem research.

Keywords: Footprint; Apogee IRT Sensor; FOV; Python Tool.

1. Background and Motivation

Apogee Instruments offers an online calculator[1] to estimate the footprint of their Infrared Radiometer (IRT) sensors. However, the tool does not provide outputs in geospatial raster formats, which are essential for integration into remote sensing workflows, GIS platforms, and environmental modeling applications.

Given the wide applicability of the IRT sensor in agricultural and forested ecosystems, we developed this Python-based tool to generate research-ready raster outputs (GeoTIFF format) of the IRT sensor's footprint. The tool enables precise georeferencing, footprint rotation (e.g., IRT looking direction and/or azimuth alignment), and compatibility with platforms such as QGIS, ArcGIS Pro, or custom scientific pipelines. This functionality not only enhances reproducibility and workflow efficiency in field-based and remote sensing research, but also serves as an educational tool to help

[1] https://www.apogeeinstruments.com/irr-calculators/

undergraduate and graduate students better understand IRT sensor behavior, footprint geometry, and geospatial data integration.

2. Research Tool Package

Figure 1 shows the modular organization of the "Apogee_IRT_Footprint" Python package. Each ".py" file represents a dedicated function or visualization component required for estimating, rotating, georeferencing, and exporting the ground footprint of the IRT sensor. The structure facilitates code readability, testing, and reuse in scientific workflows. The accompanying list highlights the core Python libraries required for executing the tool.

```
Apogee_IRT_Footprint
|
|----- __init__.py
|
|----- computer_footprint_ellipse.py
|
|----- georeference_ellipse.py
|
|----- main.ipynb
|
|----- plot_ellipse.py
|
|----- plot_ellipse_utm.py
|
|----- plot_rotated_ellipse.py
|
|----- rotated_ellipse_local.py
|
|----- save_ellipse_as_geotiff.py
|
|----- transform_ellipse_to_north.py
|
`----- tiff_to_shapefile.py
```

Required Python libraries:

1. numpy
2. matplotlib
3. rasterio
4. shapely
5. Pyproj
6. fiona

Figure 1. Directory structure and dependencies of the **Apogee_IRT_Footprint** tool.

The provided "main.ipynb" serves as the primary Jupyter notebook for executing the complete IRT footprint generation workflow, including footprint calculation, directional rotation, georeferencing, and raster export. The required inputs are not only explained in the "main.ipynb" but also in Table 1.

Table 1. Description of the input for the IRT footprint generation model.

Input name	Description
h_irt:	Height of the IRT sensor above ground (in meters).
fov_irt:	Half field-of-view angle of the sensor (in degrees, default=14°)
alpha_irt_vertical:	Tilt angle of the sensor from vertical (in degrees).
beta_irt_horizontal:	Azimuth direction of the sensor (0=north, 90=east, 180=south, 270=west, etc.)
lat:	Latitude of the IRT sensor (in WGS84).
lon:	Longitude of the IRT sensor (in WGS84).
pixel_size:	Resolution of the output raster in meters. Recommended for high pixel size.
filename:	Name of the GeoTIFF output file.

The footprint, modeled as a circle or ellipse depending on sensor angle, is first computed in a 2D Cartesian coordinate system, with the semi-major axis aligned along the x-axis. This initial shape is then rotated to align with the y-axis, simulating the default orientation facing geographic north. Next, the footprint is rotated again according to the user-defined azimuth angle ("beta_irt_horizontal"), which specifies the horizontal direction the sensor is facing (e.g., 0° = North, 90° = East). After rotation, the footprint is georeferenced to real-world coordinates based on the provided sensor location (latitude and longitude), and finally exported as a GeoTIFF image under the same package, "Apogee_IRT_Footprint."

3. Algorithm

The primary calculations are performed in the script "compute_footprint_ellipse.py." To ensure consistency with the Apogee IRT calculator, we include a screenshot (Figure 2) illustrating all variable labels used in the computation.

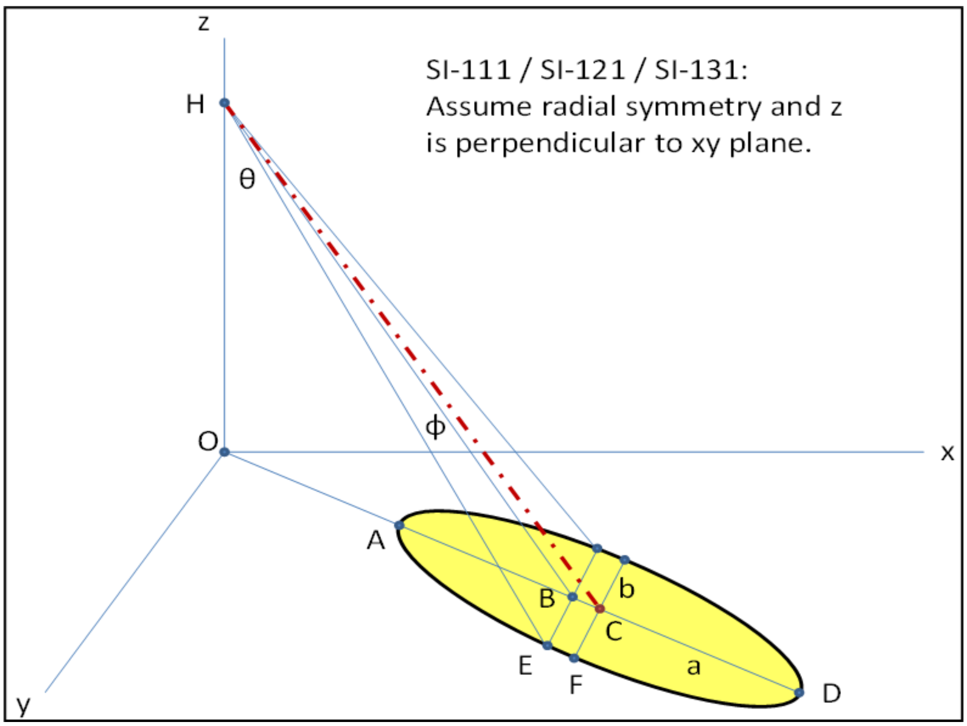

Figure 2. A screenshot from the "Excel sheet programmed to calculate target area," available at IRR Calculators[2].

The computation involves two main steps to determine the parameters of the elliptical footprint shown in Figure 2.

$$\left(\frac{BE}{CF}\right)^2 = 1 - \left(\frac{BC}{CD}\right)^2 \qquad \text{Equation 1}$$

1. Semi-major axis calculation.
   Calculated using basic trigonometric knowledge, $a = AC = DC$ will be obtained.

[2] https://www.apogeeinstruments.com/irr-calculators/

2. Semi-minor asix calculation.
   Derived using Equation 1, which defines the distance from the focus to the edge along the minor-axis direction. $b = CF$ will be obtained.

Acknowledgements

The authors are grateful for the funding support from the Secure Water Future (USDA NIFA # 2021-69012-35916) and Economic Development Administration Build Back Better Regional Challenge Farms-Food-Future Innovation Initiative (EDA #77907913). We also sincerely thank the Apogee Instruments team for their helpful responses to our questions regarding the ellipse theory used in their calculations. This tool was inspired by questions that arose during my Ph.D. research at Utah State University. I am especially grateful for the opportunity to explore this challenge in depth during my doctoral studies, which laid the foundation for the development of this tool. I sincerely appreciate the support from my supervisor and the funding, which enabled access to a real Apogee IRT sensor for hands-on experimentation. Touching it, tuning it, and exploring its behavior in the field played a key role in transforming theoretical ideas into a practical, reusable tool for the broader research community.