

¹ rushlight - Python-based Forward Modelling of ² Coronal Plasma Models

³ Sebastian Fernandes  ^{1*} and Ivan Oparin  ^{1*}

⁴ 1 Center For Solar-Terrestrial Research, New Jersey Institute of Technology, Newark, NJ 07102, USA ¶
⁵ Corresponding author * These authors contributed equally.

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review ↗](#)
- [Repository ↗](#)
- [Archive ↗](#)

Editor: [Open Journals](#) ↗

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#))

⁶ Summary

⁷ The rushlight Python package provides a framework for creating synthetic images of plasma
⁸ structures for model-to-data comparisons with coronal events. It handles the projection and
⁹ alignment of 3D simulated datasets to user-defined locations and orientations relative to the
¹⁰ sun. The produced observables are comparable to observations made by instruments such as
¹¹ the Hinode X-Ray Telescope (XRT) and the Solar Dynamics Observatory Atmospheric Imaging
¹² Assembly (AIA). rushlight aims to integrate into the growing community of Python-based
¹³ astrophysics software such as Astropy, SunPy and XRTPy.

¹⁴ Statement of need

rushlight is a Python package which performs forward modelling of simulated 3D plasma datasets in the coronal environment. Its core functionality lies in creating synthetic observables in Soft X-Ray filter bands produced by XRT, and Ultraviolet / Extreme Ultraviolet filter bands produced by AIA.

¹⁹ rushlight adapts some of the core functionality of the FORWARD package, written in the
²⁰ Interactive Data Language (IDL) ([Gibson et al., 2016](#)). It is under active development, and
²¹ aims to be continually improved as to implement more of FORWARD's features.

²² Part of rushlight's core motivation is to make EUV / SXR forward modelling more accessible
²³ to the growing company of astrophysicists who utilize the python language to develop and
²⁴ share scientific software. To this effect, rushlight has been developed as to be both compatible
²⁵ and scalable with release versions of other astrophysics open-source software, such as Astropy
²⁶ ([Astropy Collaboration et al., 2013](#)) ([Astropy Collaboration et al., 2018](#)) ([Astropy Collaboration et al., 2022](#)), SunPy ([Mumford et al., 2020](#)), and XRTPy ([Velasquez et al., 2024](#)).

²⁸ Package Structure

²⁹ rushlight's modules are organized as to promote the addition of new emission models and
³⁰ instruments to produce synthetic observables with. The package's main functionality comes
³¹ from the following classes:

- ³² ▪ `rushlight.utils.proj_imag_classified.SyntheticImage` - This module is the parent
³³ module to all other Synthetic Image classes, regardless of simulated filter type. It is
³⁴ responsible for translating user input into a single object containing both reference and
³⁵ model data. The Python module `yt` is used its ability to orient and project volumetric
³⁶ data from multiple simulation platforms.
- ³⁷ ▪ `rushlight.utils.proj_imag_classified.SyntheticFilterImage` - rushlight is in-
³⁸ tended to be expanded upon by developing other modules similar to `SyntheticFilterImage`,

39 which overloads the SyntheticImage class to apply the appropriate imaging models
40 specific to UV and SXR observations.
41 ▪ `rushlight.utils.dcube.Dcube` - This module serves to process user provided simulation
42 datasets into a YTRegion object. If one is not provided, it can generate a dummy uniform
43 grid dataset.
44 ▪ `rushlight.utils.rimage.ReferenceImage` - This module processes user provided refer-
45 ence observation maps into `sunpy.map.Map` objects from which coordinate data is later
46 calculated.
47 ▪ `rushlight.utils.synth_tools.calc_vect` - `rushlight` accepts user specification of 3
48 points in 3D space located on the intended projection plane for their simulation data.
49 From these 3 points, it uses the simulated observer's location to calculate the vector that
50 is normal to this plane, and the vector that determines the rotation of the projection
51 relative to the normal axis. These norm and north vectors, respectively, are used in the
52 `yt.off_axis_projection` module to calculate projection orientation.
53 ▪ `rushlight.utils.emission_models.uv.UVModel` - This module is used by
54 `rushlight.utils.proj_imag_classified.SyntheticFilterImage` to interpolate
55 the temperature response function for a specified AIA channel, and then to utilize the
56 density and temperature data from the simulation dataset to estimate the UV intensity
57 of the solar plasma.
58 ▪ `rushlight.utils.emission_models.xrt.XRTModel` - Similar to `rushlight.utils.emission_model`,
59 this module instead interpolates the temperature response function for a specified
60 combination of XRT filters to estimate the SXR intensity of the simulation dataset.

61 Acknowledgements

62 Test

63 References

- 64 Astropy Collaboration, Price-Whelan, A. M., Lim, P. L., Earl, N., Starkman, N., Bradley, L.,
65 Shupe, D. L., Patil, A. A., Corrales, L., Brasseur, C. E., N”othe, M., Donath, A., Tollerud,
66 E., Morris, B. M., Ginsburg, A., Vaher, E., Weaver, B. A., Tocknell, J., Jamieson, W., ...
67 Astropy Project Contributors. (2022). The Astropy Project: Sustaining and Growing a
68 Community-oriented Open-source Project and the Latest Major Release (v5.0) of the Core
69 Package. *935*(2), 167. <https://doi.org/10.3847/1538-4357/ac7c74>
- 70 Astropy Collaboration, Price-Whelan, A. M., Sipócz, B. M., Günther, H. M., Lim, P. L.,
71 Crawford, S. M., Conseil, S., Shupe, D. L., Craig, M. W., Dencheva, N., Ginsburg, A.,
72 Vand erPlas, J. T., Bradley, L. D., Pérez-Suárez, D., de Val-Borro, M., Aldcroft, T. L.,
73 Cruz, K. L., Robitaille, T. P., Tollerud, E. J., ... Astropy Contributors. (2018). The Astropy
74 Project: Building an Open-science Project and Status of the v2.0 Core Package. *156*(3),
75 123. <https://doi.org/10.3847/1538-3881/aabc4f>
- 76 Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., Greenfield, P., Droettboom, M., Bray,
77 E., Aldcroft, T., Davis, M., Ginsburg, A., Price-Whelan, A. M., Kerzendorf, W. E., Conley,
78 A., Crighton, N., Barbary, K., Muna, D., Ferguson, H., Grollier, F., Parikh, M. M., Nair, P.
79 H., ... Streicher, O. (2013). Astropy: A community Python package for astronomy. *558*,
80 A33. <https://doi.org/10.1051/0004-6361/201322068>
- 81 Gibson, S. E., Kucera, T. A., White, S. M., Dove, J. B., Fan, Y., Forland, B. C., Rachmeler,
82 L. A., Downs, C., & Reeves, K. K. (2016). FORWARD: A toolset for multiwavelength
83 coronal magnetometry. *Frontiers in Astronomy and Space Sciences, Volume 3 - 2016*.
84 <https://doi.org/10.3389/fspas.2016.00008>
- 85 Mumford, S. J., Freij, N., Christe, S., Ireland, J., Mayer, F., Hughitt, V. K., Shih, A. Y.,

- 86 Ryan, D. F., Liedtke, S., Pérez-Suárez, D., Chakraborty, P., K, V., Inglis, A., Pattnaik, P.,
87 Sipőcz, B., Sharma, R., Leonard, A., Stansby, D., Hewett, R., ... Murray, S. A. (2020).
88 SunPy: A python package for solar physics. *Journal of Open Source Software*, 5(46), 1832.
89 <https://doi.org/10.21105/joss.01832>
- 90 Velasquez, J., Murphy, N. A., Reeves, K. K., Slavin, J., Weber, M., & Barnes, W. T. (2024).
91 XRTpy: A hinode-x-ray telescope python package. *Journal of Open Source Software*,
92 9(100), 6396. <https://doi.org/10.21105/joss.06396>

DRAFT