# ADI.jl: A Julia Package for High-Contrast Imaging

Miles Lucas[1] **and Michael Bottom**[1]

1 Institute for Astronomy, University of Hawai'i

Summary

High-contrast imaging (HCI) is a powerful technique for discovering and characterizing exoplanets. Being able to probe the architecture, formation, and atmospheres of planets *directly* is necessary for advancing companion formation and evolution theory (Bowler, 2016). The process required to image an exoplanet is daunting, however, due to the brightness and proximity of their host stars. One part of making such a difficult detection is the image processing of HCI data to remove systematic signals and attenuate noise. The size of HCI data and complexity of processing algorithms requires an efficient numerical framework that simultaneously offers modularity for rapid experimentation and discovery.

Angular differential imaging (ADI) is an observational technique for HCI that utilizes the Earth's rotation throughout a night of observing (Liu, 2004; Marois et al., 2006). Normally telescopes have optics to counter this rotation, but disabling these optics and taking images throughout the night will give a sequence of frames where the sky appears to rotate. The telescope optics produce a systematic noise that will not rotate with the sky, although it can slowly vary over time. The sequence of frames are pre-processed to calibrate the images and fix defects like bad pixels. The pre-processed images are then co-aligned and concatenated together into a data cube. ADI algorithms exploit the difference in rotation to approximate and subtract the systematics without substantially overfitting the signals from potential companions.

ADI algorithms differ in how they estimate the noise model. For example instead of estimating the systematics from the *target* star, a *reference* star can be used (reference differential imaging, RDI). Instead of using the entire frame, the cube can be processed in annuli corresponding to different circumstellar regions. These geometric techniques are independent of the underlying description of the algorithms. Similarly, GPU programming or out-of-core processing are computational techniques which appear like implementation details in comparison to the algorithms or how they are applied. Creating a *modular* and *generic* framework for HCI enables scientists to explore different algorithms and techniques flexibly, allowing more thorough and deeper investigations into the capabilities of HCI for finding exoplanets.

Statement of need

`ADI.jl` is a Julia framework for post-processing high-contrast imaging (HCI) data. By organizing algorithms separately from their application, `ADI.jl` offers a modular API that benefits both observers and algorithm designers. Observers can rapidly experiment with multiple post-processing algorithms and techniques to optimally reduce their data. Julia's dynamic just-in-time (JIT) LLVM compiler (Bezanson et al., 2018, 2017) means this experimentation comes at a low runtime cost to the observer, enabling broader experimentation or higher throughput, for example, in the case of survey pipelines.

Algorithm designers will find that Julia is highly composable, so extending or adding a new algorithm only requires writing the code that is *unique to that algorithm*. Julia's language interoperability also means the algorithm can be implemented in Python or C, for example. In other words, to be able to fully use the post-processing capabilities of `ADI.jl` a new algorithm only needs to implement one or two methods. Furthermore, computational techniques like GPU programming are available *generically* through packages like `CUDA.jl` (Besard et al., 2018).

Currently `ADI.jl` supports full-frame ADI and RDI processing, with experimental support for spectral differential imaging (SDI). The algorithms that are currently implemented are median subtraction (Marois et al., 2006), principal component analysis (PCA/KLIP; Soummer et al., 2012), non-negative matrix factorization (NMF; Ren et al., 2018), and fixed-point greedy disk subtraction (GreeDS; Pairet et al., 2019, 2020). In addition, common metrics such as S/N maps and contrast curves are available for posterior analysis. Forward modeling is being built in a separate Julia package, `Firefly.jl`, as part of active research.

Comparisons with existing software

High-contrast imaging as a field predominantly utilizes Python for data reduction. We break down some of the necessary computations into *pre-processing*, which includes raw calibration of data, the centering and stacking of the data cube, bad-pixel removal, etc., *post-processing*, which includes the PSF approximation and subtraction, *detection metrics* which includes methods for analyzing post-processed data to make detections or find limits of detections, and finally *forward modeling* which includes various statistical models for companions and disks that can be used with post-processing algorithms in a maximum likelihood framework. `ADI.jl` primarily focuses on post-processing and detection metrics.

Some notable libraries for HCI tasks include the Vortex Imaging Pipeline (VIP) (Gomez Gonzalez et al., 2016), pyKLIP (Wang et al., 2015), and PynPoint (Stolker et al., 2019). A table of the feature sets of these packages alongside `ADI.jl` is presented in the online documentation. In particular, VIP has served as a useful source of information regarding HCI image-processing as well as detailed implementations of common ADI algorithms. This has been indispensable in the development of `ADI.jl`, although this package is not a direct translation. In our small benchmark suite, `ADI.jl` is roughly the same speed or slightly quicker than VIP for the algorithms we tested, except for NMF, while detection maps were ~2 orders of magnitude quicker.

In general VIP offers the most diversity in algorithms and their applications, but not all algorithms are as feature-complete as the PCA implementation. VIP also contains many useful utilities for pre-processing and a pipeline framework. pyKLIP primarily uses the PCA (KLIP) algorithm, but offers many forward modeling implementations. `PynPoint` has a highly modular pre-processing module that is focused on pipelines.

Acknowledgments

References

Besard, T., Foket, C., & De Sutter, B. (2018). Effective extensible programming: Unleashing Julia on GPUs. *IEEE Transactions on Parallel and Distributed Systems*. https://doi.org/10.1109/TPDS.2018.2872064

Bezanson, J., Bolewski, J., & Chen, J. (2018). Fast Flexible Function Dispatch in Julia. *arXiv e-Prints*, arXiv:1808.03370. http://arxiv.org/abs/1808.03370

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, *59*(1), 65–98. https://doi.org/10.1137/141000671

Bowler, B. P. (2016). Imaging Extrasolar Giant Planets. *PASP*, *128*(968), 102001. https://doi.org/10.1088/1538-3873/128/968/102001

Gomez Gonzalez, C. A., Wertz, O., Christiaens, V., Absil, O., & Mawet, D. (2016). VIP: Vortex Image Processing pipeline for high-contrast direct imaging of exoplanets. *Astrophysics Source Code Library*, ascl:1603.003. http://adsabs.harvard.edu/abs/2016ascl.soft03003G

Liu, M. C. (2004). Substructure in the Circumstellar Disk Around the Young Star AU Microscopii. *Science*, *305*(5689), 1442–1444. https://doi.org/10.1126/science.1102929

Marois, C., Lafrenière, D., Doyon, R., Macintosh, B., & Nadeau, D. (2006). Angular Differential Imaging: A Powerful High-Contrast Imaging Technique. *The Astrophysical Journal*, *641*, 556–564. https://doi.org/10.1086/500401

Pairet, B., Cantalloube, F., & Jacques, L. (2019). Reference-less algorithm for circumstellar disks imaging. *arXiv:1812.01333 [astro-Ph]*. http://arxiv.org/abs/1812.01333

Pairet, B., Cantalloube, F., & Jacques, L. (2020). MAYONNAISE: A morphological components analysis pipeline for circumstellar disks and exoplanets imaging in the near infrared. *arXiv:2008.05170 [astro-Ph]*. http://arxiv.org/abs/2008.05170

Ren, B., Pueyo, L., Zhu, G. B., Debes, J., & Duchêne, G. (2018). Non-negative Matrix Factorization: Robust Extraction of Extended Structures. *The Astrophysical Journal*, *852*, 104. https://doi.org/10.3847/1538-4357/aaa1f2

Soummer, R., Pueyo, L., & Larkin, J. (2012). Detection and Characterization of Exoplanets and Disks Using Projections on Karhunen-Loève Eigenimages. *The Astrophysical Journal Letters*, *755*, L28. https://doi.org/10.1088/2041-8205/755/2/L28

Stolker, T., Bonse, M. J., Quanz, S. P., Amara, A., Cugno, G., Bohn, A. J., & Boehle, A. (2019). PynPoint: a modular pipeline architecture for processing and analysis of high-contrast imaging data. *Astronomy & Astrophysics*, *621*, A59. https://doi.org/10.1051/0004-6361/201834136

Wang, J. J., Ruffio, J.-B., De Rosa, R. J., Aguilar, J., Wolff, S. G., & Pueyo, L. (2015). pyKLIP: PSF Subtraction for Exoplanets and Disks. *Astrophysics Source Code Library*, ascl:1506.001. https://ui.adsabs.harvard.edu/abs/2015ascl.soft06001W