Université catholique de Louvain

École polytechnique de Louvain (EPL)

Matrix Completion for Exoplanet Detection in High Contrast Imaging

Dissertation presented by
Xavier LAMBEIN

for obtaining the Master's degree in
Computer Science and Engineering

Supervisor(s)
Pierre-Antoine ABSIL, Laurent JACQUES

Reader(s)
Olivier ABSIL, Benoît PAIRET

Academic year 2017-2018

Abstract

In the past couple of decades, the discovery of planets outside our solar system has become a relatively common event. Most of them are found by indirect detection, mainly with the radial velocity and transit methods. More recently, a few extrasolar planets have been detected through direct means, by observing light produced by or reflected on the planets themselves. Unfortunately, direct observations are plagued with atmospheric and instrumental noise that decreases the maximum angular resolution we can achieve. Most of the recent advances in this field aim to find ways to eliminate this so-called "speckle noise". Angular differential imaging (ADI) offers a solution in which multiple observations of the same system are interspersed with a rotation of the field of view (FOV). The quasi-static speckle noise stays relatively constant between two observations, while the potential planets move through the FOV. With adequate post-processing, ADI can dramatically increase the detectability of planets. We introduce a new ADI post-processing technique in which data is removed around the path of a postulated planet, before being imputed back with a matrix completion method. This model is then compared to the original observations to assess whether the hypothesized planet exists. This new technique is studied with various matrix completion methods, among them a graph-regularized matrix factorization. While the results are good, they fail to improve on the state of the art. However, the idea of using a graph regularizer in low rank models without completion seems promising.

Contents

Introduction

Why look for exoplanets? As with most of research related to space, the search for exoplanets is often flagged as a somewhat useless enterprise by those who are not space-inclined. Why should the scientific community spend so much resources searching for other worlds, when the one we already have could benefit from those resources?

One answer is that space sciences do benefit us down on Earth. In addition to advancing the knowledge of humankind—arguably, a valuable goal on its own—space exploration has had a direct and measurable impact on our lives. Among many things, it brought us global telecommunications, GPS and precise weather models. The many technologies necessary to bring and sustain human life in the dangerous void of space proved useful in hospitals, households, transportation, and other parts of the daily life of humans on Earth.

Another interesting answer pertains to the awe-inspiring nature of space exploration. While the argument can be made that the Space Race was mostly another battlefront in the Cold War, it is undeniable that the human expeditions to the Moon sparked dreams in the many generations that followed, up to this day.

Those two arguments still apply to the hunt for exoplanets. Discovering worlds outside our own is of tremedous scientific importance: In the search for extra-terrestrial life, in understanding the nature and characteristics of our solar system, and even possibly in the long-term survival of our species. Furthermore, as shown by the great excitement of the space enthusiasts at each new exoplanet discovered, it also represents a great source of awe and wonder, inspiring the next generation of scientists.

How to find an exoplanet? Since the first confirmed detection in 1992 by Wolszczan and Frail, over 3,500 exoplanets have been discovered, most of them in recent years, thanks to improvements in the technology (Schneider 2018).

Until 2011, most of the detections were done with the radial velocity method. The idea behind this technique is to measure periodic shifts, called Doppler shifts, in the wavelength of the light emitted by the parent star. Those shifts are explained by the fact that the orbit of a planet around a star induces a (small) rotation of the star itself around their common center of mass. This in turn creates a periodic shift in the wavelength of the light produced by the star, due to the Doppler effect. For a review of the radial velocity method, see Lovis and Fischer (2010).

From 2012 onwards, the vast majority of detections were obtained with the transit method. This technique works by measuring the drop in brightness of a star induced by the transit of a planet in front of that star. If a planet orbits on a plane that includes the Earth, it will periodically pass in front of its star and block part of the light reaching us. By measuring this drop, it is possible to infer the existence of a planet. Deeg and Alonso (2018) present an overview of this method.

Direct imaging. This thesis is concerned with direct imaging, which only accounts for a very small percentage of detected exoplanets. Whereas the radial velocity and transit methods deduce the location of a planet by observing its effect on the parent star, direct imaging, as its name implies, seeks to directly observe light reflected or emitted by the planet itself.

This task is considerably harder than indirect detection for a couple of reasons. First, the contrast necessary to distinguish a faint planet drowned in the light of its star is difficult to achieve. Second, the observed systems are so far away from our own that planets appear very small, and can only be detected by large ground-based telescopes. At such high angular resolutions, images are overwhelmed with so-called "speckle noise", which is difficult to eliminate.

Angular Differential Imaging. A solution to the problem of speckle noise was suggested by Marois, Lafrenière, Doyon, et al. (2006) in a technique called Angular Differential Imaging (ADI). This method achieves noise reduction by taking advantage of the rotation of the Earth, and the fact that the speckle noise is relatively static. During a single night of observation, photographs of a specific star system are taken at a regular interval, forming a kind of video. Because the Earth rotates slightly between two pictures, the planets orbitting a star will appear to rotate around that star, while the speckle noise, originating in the instruments, will roughly stay in place. By de-rotating and combining the images of one night, it becomes possible to significantly increase the signal-to-noise ratio.

Our solution. The work developed in this thesis is built upon ADI. Like many others, our technique builds a low-rank model of the star and speckle present in the ADI "video" (hereafter referred to as "data cube"). The technique stands out in how this model is constructed: First, we postulate the existence of a planet at a given location, and compute its path through the data cube. Then, we remove this path from the cube and use matrix completion to fill in the missing data. Finally, we compute the intensity (astronomical flux) of the postulated planet with a maximum likelihood estimator that compares the model with the original data cube along the path of the planet. If the model resembles the data, we are unlikely to find a planet there. Conversely, if there is a discrepancy between the two, it might be explained by the presence of an exoplanet.

Structure of this document. The next chapter is a summary of the state of the art in direct imaging, ADI post-processing and matrix completion. Chapter 2 characterizes the data used for the solution presented here, and introduces the VORTEX Image Processing toolbox for direct imaging. In chapter 3, we rigorously describe our proposed solution for exoplanet detection in an ADI context. Chapter 4 explains the metrics used for assessing the quality of our solution, and the next chapter presents and discusses the results obtained against these metrics. The final chapter concludes and lists potential directions for further research.

Chapter 1

State of the Art

1.1 Direct Imaging of Exoplanets

Direct imaging of exoplanets is a challenging task. Stars are orders of magnitude brighter than the planets orbitting them, their signal overwhelms that of their surroundings. Most extrasolar systems are so far away that planets often appear separated from their parent star by only a fraction of an arcsecond. Such a high resolution is currently impossible to achieve from space, and ground-based imaging is plagued with atmospheric noise. Even when these issues have been addressed, imperfections in the instuments produce patterns that are visually similar to and sometimes brighter than the actual planets we are attempting to find.

In this section, we briefly cover the imaging techniques that allow astronomers to produce the kind of high-angular resolution, high-contrast observations necessary for the direct detection of exoplanets. Most of what follows is based on the work of Duchêne in "High-angular resolution imaging of disks and planets". For a more in-depth discussion of this subject, we refer the reader to Duchêne (2008).

The coronagraph. In order to reduce the magnitude between the host star and orbitting objects, telescopes are equiped with a *coronagraph*. A coronagraph is a device designed to block on-axis light, while allowing off-axis light to come through relatively unchanged. The classical design was introduced by Lyot to study the corona of the Sun (Lyot 1939). Its main components are an opaque mask on the focal plane blocking on-axis light, and an undersized pupil (the "Lyot mask") blocking diffracted starlight (see Fig. 1.1).

The performance of coronagraphs is measured in terms of *contrast*, which is the ratio of the

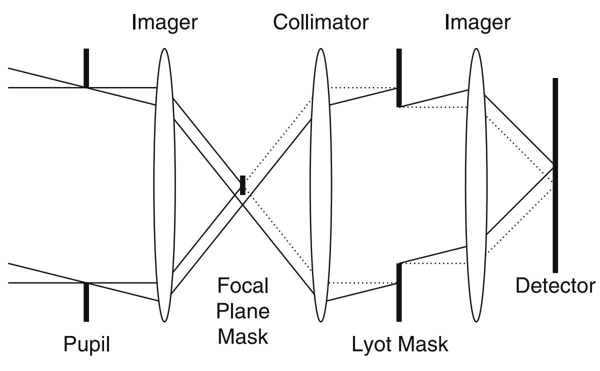

Figure 1.1: Components of the traditional Lyot coronagraph. A focal plane mask blocks direct on-axis starlight, while the Lyot mask blocks diffracted starlight. Credit: Oswalt and McLean (2013).

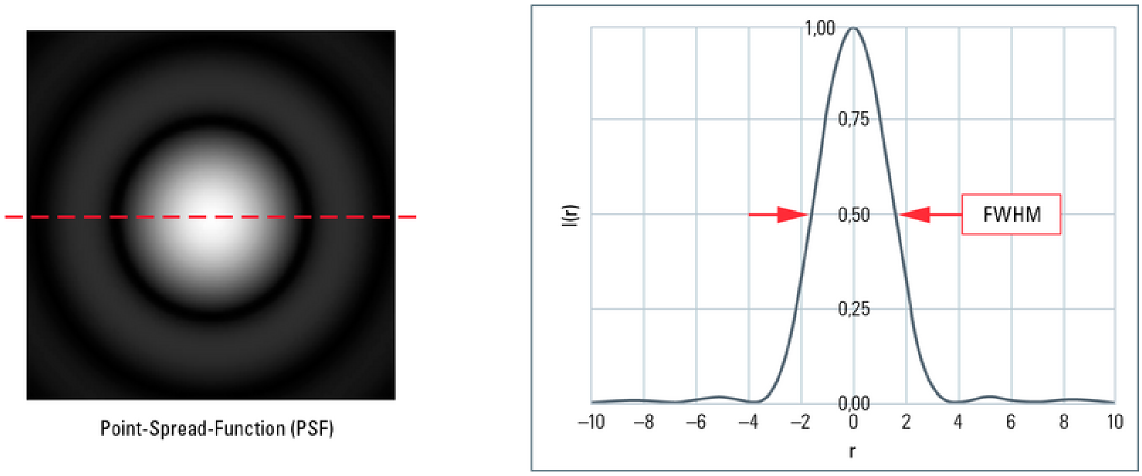

Figure 1.2: Left: Airy pattern produced by a point source. Right: Cut of the diffraction pattern through the center to show the full width half maximum. Credit: Borlinghaus (2015).

minimum detectable flux to the on-axis intensity if it were not occulted (Oswalt and McLean 2013). While modern coronagraphs can achieve impressive contrasts, most planets are still too dim to be detected directly. Hence, direct detection is mainly concerned with young, hot planets, which are still emitting plenty of light in the infra-red spectrum.

Ground-based observation. At a fundamental level, telescopes are limited in their resolution by the diffraction of light. Light from a point source at infinity, reflected on the mirror of a telescope, creates on the screen a pattern made of a single bright spot in the middle and a series of darker concentric circles (Fig. 1.2, left). Such a pattern is called an *Airy pattern*, and its size is measured at the full width at half-maximum (FWHM) (Fig. 1.2, right).

When observing a distant scene—say, a planetary system—this effect can be modeled as a convolution of the scene by the Airy pattern. Hence, this pattern acts as the impulse response of the telescope, and is commonly referred to as *point spread function* (PSF). The presence of Airy patterns limits the angular resolution of a telescope, because details sensibly smaller than the FWHM blend together into a single spot, rendering them indistinguishable.

For a given telescope of diameter D observing light at wavelength λ (both measured in meters), the FWHM of Airy patterns is roughly $\theta_{\text{diff}} = \lambda/D$ radians. An exoplanet 10 pc away orbitting its star at 1 AU will have an angular separation of $0.1'' = 4.85 \times 10^{-7}$ rad (Traub and Oppenheimer 2010). If we want to have a chance of distinguishing it from the host star in, say, the L'-band infrared ($\lambda = 3.8\,\mu\text{m}$), our telescope needs to have a diameter of at least:

$$D = \frac{\lambda}{\theta} = \frac{3.8 \times 10^{-6}\,\text{m}}{4.85 \times 10^{-7}} \approx 10\,\text{m}.$$

Given the current state of technology, such a large class of telescopes can only be found on the ground. In comparison, the largest active space telescope is the Hubble Space Telescope, which has a diameter of only 2.4 m.

The disadvantage that comes with ground-based observations is that one has to deal with perturbations in the light introduced by the atmosphere. For all intents and purposes, light from a distant star arrives to us as a plane wave. However, when going through the atmosphere, this wave encounters variations in the index of refraction that distort the wavefront, introducing so-called *speckle noise* in the final image. Realizations of this noise are very short-lasting, so much so that over a second, individual speckles sum up to a large Gaussian halo, whose FWHM is called *seeing*, written θ_{atm}. The seeing sets a lower bound on the effective angular resolution

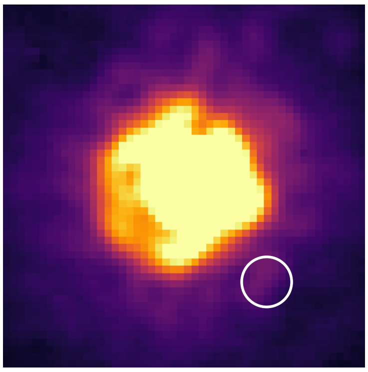

Figure 1.3: An AO image of the system β Pictoris, taken by the VLT/NACO telescope. High values in the center have been thresholded to make the speckles on the outskirts visible. The location of a companion planet is circled, but it is drowned in the surrounding speckle noise.

of any ground-based telescope, no matter their diameter. Unfortunately, θ_{atm} typically ranges from $0.5''$ to several arcseconds, which is much too high for direct imaging.

Adaptive optics. Several solutions exist to counter the effect of the atmosphere and achieve an effective resolution closer to the diffraction limit. *Speckle interferometry* (Saha 2003) takes many short-exposure pictures (10–100 ms)—such that individuals speckles don't have time to create a seeing—and computes the autocorrelation of those images. This eliminates the decorrelated speckles, while preserving the signal of interest. However, this technique works poorly in high contrast imaging because faint objects do not emit enough photons to be detected in a short exposure time.

Adaptive optics (AO) provides another solution to the problem of atmospheric speckles. The idea of AO is to use a *deformable mirror* that can adapt in real time to the current wavefront, so as to correct for the atmosphere-induced distortions. In this setting, it becomes possible to integrate over a much longer time, collecting enough photons for the purpose of HCI. For a summary of AO see Beckers (1993).

Differential techniques. Even with good AO, the actual impulse response of a telescope never truly looks like a perfect Airy pattern: Imperfections in the instruments introduce additional *static* or *quasi-static speckles* in the PSF. They appear in as spots of size λ/D, which look inconveniently like the companions we are trying to find (see Fig. 1.3). These speckles have a much longer lifespan, from a minute to a few hours, making them impossible to "average out" across several pictures. Instead, more sophisticated, so-called *differential* techniques, coupled with adequate post-processing, are required (Duchêne 2008).

The idea behind a differential technique is to introduce diversity in the data by taking multiple pictures with the same realization of the quasi-static speckle noise, but with different "realizations" of potential companions, such that it becomes possible to separate the noise from the exoplanets. *Simultaneous Differential Imaging* (SDI), first introduced by Racine et al. (1999), takes advantage of the fact that the radial location of the speckles depends linearly on the wavelength λ, while the location of companions does not. The telescope takes a single picture and runs it through several adjacent passband filters. Those images are then radially scaled, such that any companion finds itself at a different location on each image, while the speckles are now aligned. In *Polarimetric Differential Imaging* (PDI), proposed by Kuhn et al. (2001), two pictures are taken with orthogonal polarizing filters. The starlight and speckles are assumed to

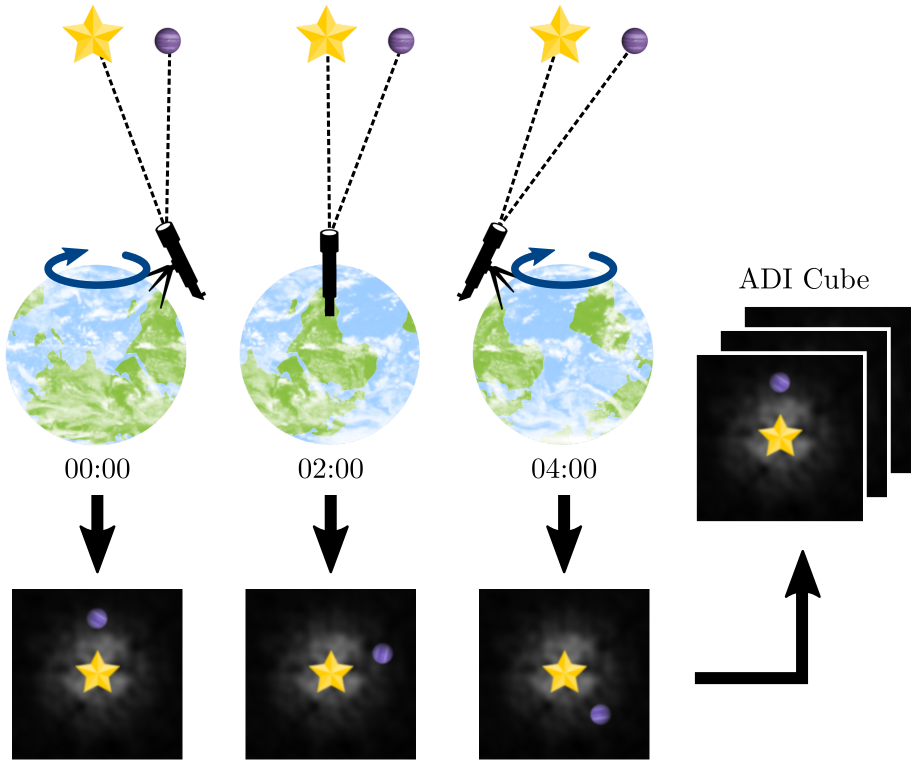

Figure 1.4: Visualization of the ADI observation process. A telescope pointed at a distant star takes a series of pictures over the course of a single night. The rotation of the Earth makes companions follow a circular trajectory in successive observations. The stack of individual frames create a video called ADI cube. While only three frames are depicted here, real cubes typically contain up to a few hundreds. Usually, the planets are much too faint to be visible on individual frames.

be unpolarized, and will cancel out when subtracting one picture from the other, revealing the (polarized) light scattered off objects around the star.

A third and very common technique is *Angular Differential Imaging* (ADI), which is the basis for the work in this thesis. It was introduced by Marois, Lafrenière, Doyon, et al. (2006). In ADI, diversity is introduced by taking several pictures over a single night of observation while letting the field of view (FOV) rotate at a known rate. Since only the FOV—and not the PSF of the telescope—rotates with time, the speckle noise remains roughly constant between two consecutive images, while any potential companion is smeared azimuthally (see Fig. 1.4). The result is a "video", called *ADI cube*, composed of many frames containing the star, the quasi-static speckle, and companions following a circular trajectory.

1.2 Post-Processing Techniques in ADI

Since the introduction of ADI, a large number of post-processing techniques have been developed to reveal the companions present in a data cube and perform detection. Most of these techniques aim to create a model of the star and speckle noise, the so-called *model PSF*. This model takes the shape of an ADI cube which, when subtracted from the orginial data, (partially) removes the star and noise. The goal is for the resulting *residual cube* to contain as much planetary signal as possible, while limiting the residual noise.

To amplify the signal-to-noise ratio of companions, the residual cube is de-rotated such that points along circular paths (*i.e.* companions) become aligned. Finally, the frames are

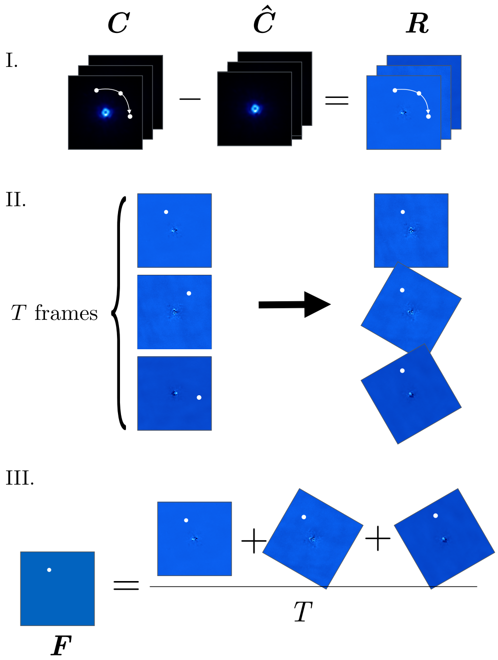

Figure 1.5: Representation of the ADI post-processing. $\boldsymbol{C}$ is the data cube, $\hat{\boldsymbol{C}}$ is the model PSF and $\boldsymbol{R}$ is the residual cube. T is the number of frames in $\boldsymbol{C}$, and $\boldsymbol{F}$ is the processed frame. Credit: Pairet et al. (2018).

mean-combined to produce a single image, called *processed frame*. When this happens, the planetary signals are added up coherently, while any remaining noise is averaged out. This entire process is represented on Fig. 1.5.

In the following, we will present several techniques commonly used to produce a processed frame. The first three, c-ADI, PCA and LLSG, follow the scheme of the previous paragraph and produce a model PSF representing the star and noise. The fourth, called ANDROMEDA, is slightly different. It seeks to generate a processed frame directly by maximizing the likelihood of observing a companion along a given trajectory in the cube.

1.2.1 Classical ADI

The first technique based on ADI is called *classical ADI* (c-ADI). It was presented in the same paper that introduced the ADI method of observation (Marois, Lafrenière, Doyon, et al. 2006).

Given a data cube $\boldsymbol{C}$, c-ADI builds a global model PSF by computing the median of $\boldsymbol{C}$ along the time dimension. Because many of the speckles have a lifespan shorter than the "duration" of the dataset, this first step is typically not enough to remove them. In a second step, a model PSF is constructed for each individual frame $\boldsymbol{C}(t)$ by taking the median of four frames $\boldsymbol{C}(t-b), \boldsymbol{C}(t-b-1), \boldsymbol{C}(t+c), \boldsymbol{C}(t+c+1)$, with b and c chosen such that there is at least $1.5\lambda/D$ FOV orientation difference between frame $\boldsymbol{C}(t)$ and frames $\boldsymbol{C}(t-b)$ and $\boldsymbol{C}(t+c)$. Since this requirement depends on the separation from the center, the frame is first split into annuli, and the process is applied independently on each annulus.

1.2.2 Principal Component Analysis

Principal component analysis (PCA) is a classical tool in data analysis used to extract the general trends of a dataset. Given a collection of vectors, PCA projects them onto an orthogonal basis

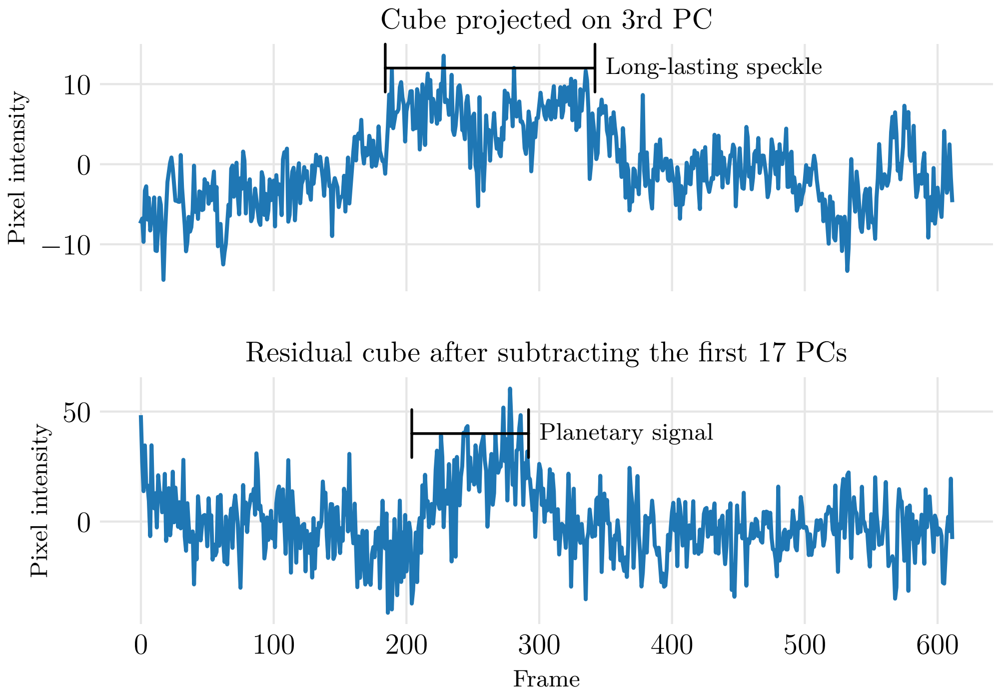

Figure 1.6: Top: Intensity of a single pixel on the trajectory of an exoplanet, in the cube projected onto the 3rd principal component. Bottom: Intensity of the same pixel in the residual cube, after removal of the first 17 principal components. The planet goes through that pixel around frames 200-300.

in which the data becomes uncorrelated, and most of its variance ends up lying along the first few basis vectors, called *principal components* (see *e.g.* Jolliffe 2002).

The idea of applying PCA to PSF estimation was introduced simultaneously under the names "PYNPOINT" (Amara and Quanz 2012) and "KLIP", as a projection onto a Karhunen-Loève basis (Soummer et al. 2012). In this context, the vectors are the frames of the data cube: Each individual N-by-N image can be "vectorized" by stacking its columns into a long vector of size N^2. This collapses the cube $\boldsymbol{C} \in \mathbb{R}^{T\times N\times N}$ into a matrix $\boldsymbol{M} \in \mathbb{R}^{T\times N^2}$ in which rows are frames and columns are pixels. Applying PCA to this matrix reveals its underlying basis $\boldsymbol{U}$ of "principal frames".

The assumption is that the star and long-lasting speckles should be part of the "general trend" of the dataset, and thus they will be absorbed in the first few principal components. The faint, moving signal of the companions, on the other hand, is thought to be higher-rank, requiring many more components to be fully expressed. Hence, for a well-chosen rank k, projecting $\boldsymbol{M}$ onto the first k vectors of $\boldsymbol{U}$, written $\boldsymbol{U}_k$, should produce a good model PSF. This is indeed what is observed in practice, as illustrated in Fig. 1.6, where a single pixel, projected onto the third component, captures a long-lasting speckle. That same pixel, when projected onto $\operatorname{span}(\boldsymbol{U}) - \operatorname{span}(\boldsymbol{U}_{17})$, shows the signal of the planet passing through.

There are many ways of performing PCA on a matrix. The most common is to apply *singular value decomposition* (SVD), in which a matrix $\boldsymbol{M} \in \mathbb{R}^{m\times n}$ of rank r is factorized into:

$$\boldsymbol{M} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top, \tag{1.1}$$

where $\boldsymbol{\Sigma} \in \mathbb{R}^{r\times r}$ is the diagonal matrix of ordered singular values, and $\boldsymbol{U} \in \mathbb{R}^{m\times r}$ and $\boldsymbol{V} \in \mathbb{R}^{n\times r}$ are orthonormal matrices. It can be shown (again, *e.g.* Jolliffe 2002) that, in this decomposition, $\boldsymbol{U}$ is exactly the (ordered) principal basis, and $\boldsymbol{U}^\top\boldsymbol{M} = \boldsymbol{\Sigma}\boldsymbol{V}^\top$ are the coefficients of the vectors

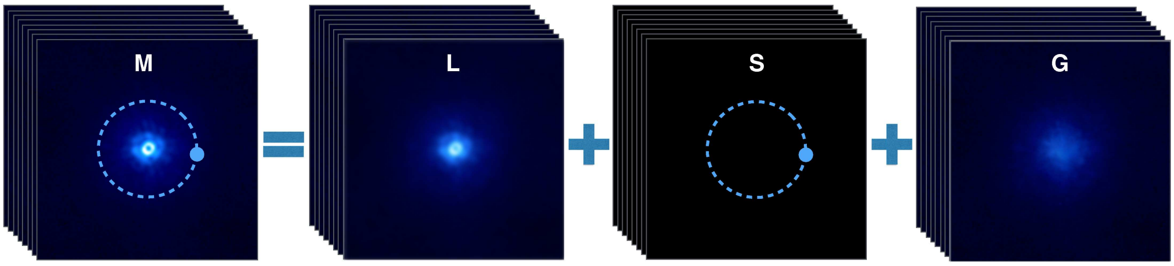

Figure 1.7: LLSG decomposition of an ADI data cube (*left*) into low-rank (*middle left*) plus sparse (*middle right*) plus Gaussian noise (*right*) terms. Credit: Gonzalez, Absil, et al. (2016).

of $\boldsymbol{M}$ expressed in this basis. The projection onto the first k principal components can be seen as a truncated version of SVD:

$$\boldsymbol{U}_k \boldsymbol{U}_k^\top \boldsymbol{M} = \boldsymbol{U}_k \boldsymbol{U}_k^\top \boldsymbol{U} \boldsymbol{\Sigma} \boldsymbol{V}^\top = \boldsymbol{U}_k \boldsymbol{\Sigma} \boldsymbol{V}^\top = \boldsymbol{U}_k \boldsymbol{\Sigma}_k \boldsymbol{V}_k^\top . \tag{1.2}$$

PCA is a simple yet very efficient way of producing a good model PSF. Its main disadvantage is that a large proportion of the planetary signal is still included in the first k principal components, leading to an underestimate of the flux. This effect is exacerbated at closer radii because planets becomes more correlated between frames.

1.2.3 LLSG

PCA can be seen as splitting the data cube into a low-rank part and a noise part. The *Local Low-rank plus Sparse plus Gaussian-noise* (LLSG) decomposition, introduced by Gonzalez, Absil, et al. (2016), attempts a separation into three parts: A low-rank term that absorbs most of the star and speckle, a sparse term that absorbs the planetary signals, and a residual Gaussian noise term (see Fig. 1.7).

LLSG is based on the GoDec algorithm ("Go Decomposition", Zhou and Tao 2011; Zhou and Tao 2013), which performs a three-terms decomposition that minimizes:

$$\begin{aligned} \min_{\boldsymbol{L},\boldsymbol{S}} \quad & \|\boldsymbol{M} - \boldsymbol{L} - \boldsymbol{S}\|_F^2 + \lambda \|\mathbf{vec}(\boldsymbol{S})\|_1 \\ \text{subject to} \quad & \operatorname{rank}(\boldsymbol{L}) \leq k, \end{aligned} \tag{1.3}$$

where $\boldsymbol{M}$ is the data cube in matrix form (as introduced with PCA), $\boldsymbol{L}$ and $\boldsymbol{S}$ are the low-rank and sparse terms, λ is a soft threshold to select, and $\mathbf{vec}(\cdot)$ is the operator that flattens a matrix into a column vector.

This problem is solved with alternate minimization on $\boldsymbol{L}$ by hard-thresholding the singular values of $\boldsymbol{M} - \boldsymbol{S}$, and on $\boldsymbol{S}$ by soft-thresholding the entries of $\boldsymbol{M} - \boldsymbol{L}$. See section 1.4.2 for more details on singular value thresholding.

The rank k is a critical parameter and must be hand-picked, but the soft threshold λ for $\boldsymbol{S}$ is selected automatically with a local heuristic. The cube is split into patches and a different threshold is computed for each of them by taking the square root of median absolute deviation of the entries of $\boldsymbol{S}$ in that patch.

1.2.4 ANDROMEDA

Unlike the previous methods, ANDROMEDA does not seek to create a model PSF and generate a processed frame. Instead, this technique defines a stochastic model of the data parameterized by the location and astronomical flux of the companions. In this context, maximum likelihood estimation can be used to find the exoplanets. This method was introduced by Cantalloube et al. (2015).

ANDROMEDA starts with a pre-processing step where the data cube is run through a high-pass filter. This removes the lower spatial frequencies, improving the "Gaussianity" of the noise.

Once this is done, a set of *differential frames* $\boldsymbol{\Delta}(k)$ is constructed, each of them being the difference of two frames $\boldsymbol{C}(t_k)$ and $\boldsymbol{C}(t'_k)$ of the (pre-processed) ADI cube. The pairs of frames are selected as close in time as possible, while having enough azimuthal rotation between them to avoid companion self-subtraction. Since this condition depends on the distance from the center, the cube is first split into annuli, and a different model is constructed for each of them.

The following process is repeated for every trajectory in the cube—that is, for every possible starting point $\boldsymbol{r}_0 \in \mathbb{N}^2$ in the first frame. Assuming this trajectory contains a companion, the differential frames can be modeled as:

$$\boldsymbol{\Delta}(k) = a\boldsymbol{P}_{\boldsymbol{r}_0}(k) + \boldsymbol{N}(k), \tag{1.4}$$

where a is the flux of the postulated planet, $\boldsymbol{P}_{\boldsymbol{r}_0}$ is the planet signature in differential frames, and $\boldsymbol{N}$ is the residual noise. More specifically, $\boldsymbol{P}_{\boldsymbol{r}_0}(k)$ is the sum of two reference planetary signatures (reference PSFs, see section 2.2) of opposite signs. These signatures are placed so as to correspond to the location of the postulated planet in frames $\boldsymbol{C}(t_k)$ and $\boldsymbol{C}(t'_k)$.

Under the hypothesis that $\boldsymbol{N}$ is white, Gaussian and non-stationary, the likelihood is given by:

$$L_{\boldsymbol{r}_0}(a) \propto \exp\left(-\frac{1}{2}\sum_k \sum_{\boldsymbol{r}} \frac{|\boldsymbol{\Delta}(k,\boldsymbol{r}) - a\boldsymbol{P}_{\boldsymbol{r}_0}(k,\boldsymbol{r})|^2}{\sigma^2_{\boldsymbol{\Delta}}(\boldsymbol{r})}\right), \tag{1.5}$$

where the inner sum iterates over pixels, and $\sigma^2_{\boldsymbol{\Delta}}$ is the variance (empirical, across k) of the differential frames. While speckle noise itself is not Gaussian—the distribution is closer to a *modified Rician* (Fitzgerald and Graham 2006)—the high-pass filtering and the difference of frames have a whitening effect, making this assumption more realistic. The most likely planet intensity $\hat{a}_{\boldsymbol{r}_0}$ is found when maximizing $L_{\boldsymbol{r}_0}(a)$, or alternatively, its logarithm:

$$L_{\boldsymbol{r}_0}(a) \propto -\frac{1}{2}\sum_k \sum_{\boldsymbol{r}} \frac{|\boldsymbol{\Delta}(k,\boldsymbol{r}) - a\boldsymbol{P}_{\boldsymbol{r}_0}(k,\boldsymbol{r})|^2}{\sigma^2_{\boldsymbol{\Delta}}(\boldsymbol{r})}, \tag{1.6}$$

which gives the following closed-form solution:

$$\hat{a}_{\boldsymbol{r}_0} = \operatorname*{argmin}_a \log L_{\boldsymbol{r}_0}(a) = \frac{\sum_{k,\boldsymbol{r}} \boldsymbol{P}_{\boldsymbol{r}_0}(k,\boldsymbol{r})\boldsymbol{\Delta}(k,\boldsymbol{r})/\sigma^2_{\boldsymbol{\Delta}}(\boldsymbol{r})}{\sum_{k,\boldsymbol{r}} \boldsymbol{P}^2_{\boldsymbol{r}_0}(k,\boldsymbol{r})/\sigma^2_{\boldsymbol{\Delta}}(\boldsymbol{r})}. \tag{1.7}$$

It can be shown, under the strong aforementioned hypothesis on the noise, that the log-likelihood as described in (1.6) represents the square of the *signal-to-noise ratio* (SNR), which can be used to perform detection (see section 1.3). However, in practice, because the noise is never truly Gaussian, an additional post-processing step is required, in which the SNR map is normalized by its own empirical standard deviation.

1.2.5 Wavelet-based Speckle Suppression

Recently, Bonse et al. (2018) improved remarkably on PCA by pre-processing the cube with a wavelet-based denoising step. Like a Fourier transform, the wavelet transform is a decomposition of a signal into frequencies. Unlike Fourier however, it is localized in time, revealing the spectrum around a specific instant.

To a given pixel $\boldsymbol{r}$ corresponds a time series $\boldsymbol{p}_{\boldsymbol{r}}(t) = \boldsymbol{C}(t,\boldsymbol{r})$ which lists the values taken by that pixel on each frame t. As time goes, the planets move through the cube. If no companion

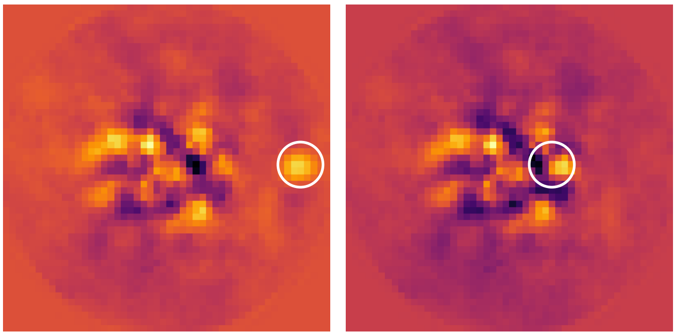

Figure 1.8: Comparison of two processed frames with an injected fake companion at distances $4.5\,\lambda/D$ (*left*) and $1.5\,\lambda/D$ (*right*) from the center. While the companions display similar intensities in the two frames, the one closer to the edge can be easily distinguished from the background noise, while the one near the center cannot.

passes through $\boldsymbol{p_r}$, then the series will be noisy but relatively flat. If a planet does pass through it, a slow increase and drop of flux is expected.

The proposed pre-processing step goes as follows. A wavelet transform is applied to each pixel, revealing which frequencies are activated around which instants. This time-frequency space can be exploited to separate the planetary signals from the noise. Because planets slowly enter and exit a pixel, they tend to be localized in the lower frequencies, whereas noise is more spread across the spectrum. Denoising is performed with *wavelet shrinkage*, in which coefficients are soft-thresholded with a value scaled by the estimated noise levels (Dohono 1992). Reconstruction from the thresholded wavelet space gives the denoised time series $\boldsymbol{p_r}(t)$.

1.3 Performing Detection

While it is possible to find companions directly in the processed frame, either with the naked eye or by thresholding with some minimum flux, performing detection in this way provides no guarantee that a spot of bright pixels is indeed a companion and not part of the residual noise. Because the variance of this noise gets smaller with increasing radius, a spot of medium intensity in the outskirts of the frame is more likely to correspond to a companion than the same spot near the center (see Fig. 1.8).

1.3.1 Hypothesis Testing

Formally, detection is expressed in terms of hypothesis testing. Let $\boldsymbol{X}$ be a patch of pixels in the processed frame. Two opposing hypotheses are being considered: The *null hypothesis* H_0 states that $\boldsymbol{X}$ only contains residual noise, and the *research hypothesis* H_1 states that it also contains a companion. In that context, $\boldsymbol{X}$ is written as:

$$\boldsymbol{X} = a\boldsymbol{P} + \boldsymbol{N}, \tag{1.8}$$

where $\boldsymbol{P}$ is the planetary signature, a is its flux, and $\boldsymbol{N}$ is the residual noise. When H_0 is true, $a = 0$; otherwise, $a > 0$.

Since there is no prior knowledge on a, H_1 is accepted by rejecting H_0—that is, by showing that $\boldsymbol{X}$ takes values above those which are reasonably explained by the distribution of $\boldsymbol{N}$. What "reasonably" actually means depends on the chosen *confidence level* (CL). A typical choice in

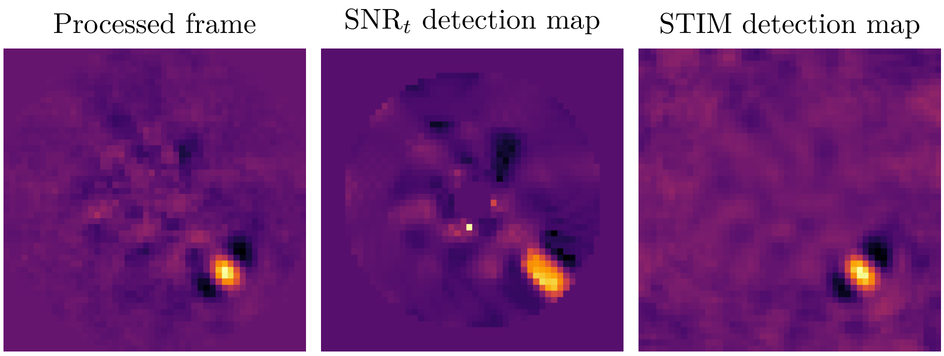

Figure 1.9: Left: Processed frame after performing PCA on the VLT/NACO β Pictoris ADI cube. Middle: Corresponding SNR_t detection map. Right: Corresponding STIM detection map.

astronomy is to pick a CL that corresponds to a 5σ separation from the mean of a Gaussian distribution, giving $\mathrm{CL} \approx 1 - 3 \times 10^{-7}$ (see *e.g.* Marois, Lafrenière, Macintosh, et al. 2008).

To perform detection on the processed frame with a given CL, one needs to construct a model of the noise $\boldsymbol{N}$ and assess whether $\Pr(\boldsymbol{N} \geq \boldsymbol{X}) \leq 1 - \mathrm{CL}$. This test is summarized within a *test statistic* T, which conveys information on the likelihood of H_0.

By computing the test statistic for every patch on the processed frame, a *detection map* is created. The detection map is a heatmap showing where one is likely to find a companion. Based on the distribution of T, this heatmap can be thresholded with a value τ corresponding to the objective CL. Then, detection is a matter of finding adjacent non-zero pixels on the thresholded map. Fig. 1.9 shows an example of a processed frame, as well as the two corresponding detection maps introduced below.

1.3.2 The SNR Detection Map

A common statistical test is the single-sample Z-test (see *e.g.* Sprinthall 2011), which supposes normally distributed samples under H_0. This corresponds to $\boldsymbol{N}$ being a Gaussian noise.

Under the null hypothesis, the mean of $\boldsymbol{X}$ is zero, and the Z statistic reduces to computing the empirical mean-to-variance ratio:

$$Z = \mathrm{SNR} = \frac{\mu}{\sigma}. \tag{1.9}$$

In astronomy, this quantity is usually called *signal-to-noise ratio* (SNR) (Schroeder 2000).

Traditionally, the SNR was computed as follows. For a given pixel, $\boldsymbol{X}$ is the disk of diameter λ/D centered around that pixel. The mean μ is computed over $\boldsymbol{X}$, while the variance σ^2 is estimated over an annulus of width λ/D (see Fig. 1.10). This way, σ takes into account the fact that the noise distribution varies with the distance from the star.

However, this definition of the SNR suffers from multiple problems. First, the distribution of $\boldsymbol{N}$ is not actually Gaussian. Instead, it was shown to follow a *modified Rician* (Fitzgerald and Graham 2006), which has a heavier tail. Consequently, the Z-test does not standardize the noise, and the CL is overestimated. In fact, Marois, Lafrenière, Macintosh, et al. (2008) showed that the real CL can be smaller by several orders of magnitude.

Second, this SNR assumes independent pixels when estimating the variance, but $\boldsymbol{N}$, being the result of AO speckles, is actually correlated in space. Therefore, the number of independent samples is not the number of pixels in the annulus: Rather, it is the number of non-overlapping resolution elements of size λ/D at that radius. At a distance $r \times \lambda/D$ from the center, there are $E_r = \lfloor 2\pi r \rfloor$ such elements (see Fig. 1.11). This effect radically decreases the number of samples available, especially at smaller radii.

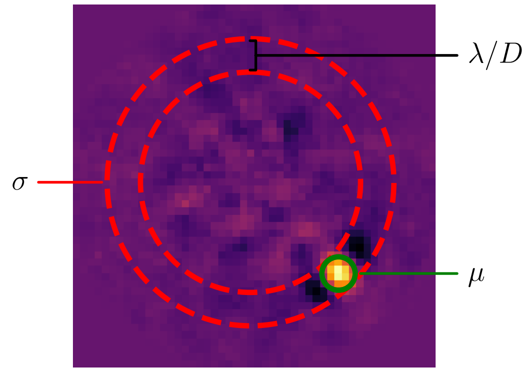

Figure 1.10: Illustration of how the mean and variance are estimated in the computation of the SNR at a given location (here, the pixel at the center of the small circle). This figure shows a processed frame after performing PCA on an ADI cube. The small continuous circle represents the area over which the mean μ is estimated. The large dashed annulus (excluding the circle) represents the area over which the standard deviation σ is estimated.

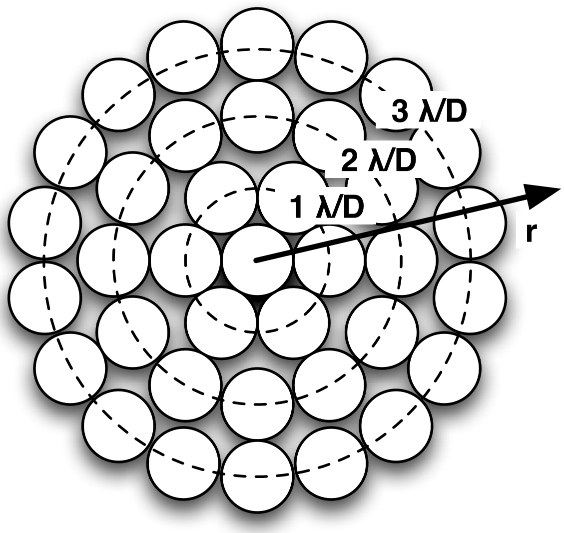

Figure 1.11: Number of distinct resolution elements as a function of the radius (expressed in multiples of λ/D). Credit: Mawet et al. (2014).

To address these issues Mawet et al. (2014) proposed to replace the Z-test with a two-sample Student t-test. The t-test was designed to take into account the small sample statistics, and was shown to be more robust to deviations from a normal distribution. The two-sample t-test verifies whether two samples of n_1 and n_2 independent elements are drawn from the same population of unknown mean and variance.

In the case of detection, one resolution element is compared against the $E_r - 1$ other at the same radius. Under H_0, those two samples should come from the same parent distribution. This can be formulated as a two-sample t-test in which the first sample contains one resolution element, while the other sample contains the rest.

Let x_1 be the flux of an element. Let $\overline{x}_2$ and s_2 be the average flux and standard deviation of the remaining $E_r - 1$. The SNR_t is:

$$\mathrm{SNR}_t = \frac{x_1 - \overline{x}_2}{s_2\sqrt{1 + \frac{1}{E_r - 1}}}, \tag{1.10}$$

where SNR_t stands for "SNR with t-test". This statistic can be used for robust detections.

1.3.3 The STIM Detection Map

The SNR_t, while more robust than the SNR, still suffers from multiple problems. While the t-test takes into account the small-sample statistics, it only quantifies the effect without addressing it. This dramatically reduces the CL achievable near the star. The SNR_t is also not robust to the presence of more than one companion in the same annulus. If two companions exist at a close radius, $\overline{x}_2$ and s_2 will be overestimated, leading to a smaller SNR_t.

To address these problems, Pairet et al. (2018) suggested a new detection map, the *standardised trajectory intensity mean* (STIM) map, which is computed on the residual cube $\boldsymbol{R}$, rather than the processed frame. For a given trajectory g, the detection value is computed as:

$$d_g = \frac{\mu_g}{\sigma_g}, \tag{1.11}$$

where μ_g and σ_g are the empirical mean and standard deviation of $\boldsymbol{R}$ along g. The idea is that, under H_0, the mean μ_g should be close to zero, while the variance should be high, giving $d_g \approx 0$. Under H_1, the converse is expected to be true, giving $d_g \gg 0$.

This detection map is not impacted by the small-sample statistics because the mean and variance are computed on the number of frames, which is the same for each trajectory. In practice, this is not entirely true because trajectories are more correlated at smaller radii. However, Pairet et al. (2018) quantified this effect and showed that, with enough frames, it quickly becomes negligible.

The second problem is adressed too: Because d_g is computed for one trajectory independently from the others, its value will not be underestimated if two planets coexist in the same annulus.

The STIM detection map is also faster to compute than the SNR_t map. Trajectories appear as straight lines in the derotated cube, so one only has to compute the temporal mean and variance on the residual cube after derotation, and to take their ratio entry-by-entry.

One drawback of this method is that the test statistic does not provide a way to select a threshold from a given CL. Pairet et al. (2018) address this by estimating the noise level on the cube rotated in the opposite direction. Its STIM map will have a similar distribution to that of the original dataset, but without the planetary signals.

1.4 Matrix Completion

Matrix completion is the problem of recovering all the entries of a matrix $\boldsymbol{M} \in \mathbb{R}^{m \times n}$ by observing only a subset $\Omega \subset \mathbb{N}^2$ of them. Without any prior on $\boldsymbol{M}$, this problem is impossible to solve—it is analoguous to guessing random numbers. A possible hypothesis is to assume that the matrix has (approximately) *low-rank*—that is, has rank $r \ll \min(m, n)$. This simpler problem, *low-rank matrix recovery*, was shown to have an exact solution under some conditions on the structure of $\boldsymbol{M}$ and Ω (E. J. Candes and Recht 2008).

This amazing result is explained by the fact that an m-by-n matrix of rank r has mn entries, but only $(m + n - r)r$ degrees of freedom, corresponding to those of its left and right singular vectors and of its singular values:

$$\mathrm{DOF}_{\mathrm{left}} = \sum_{i=1}^{r}(m - i) = mr - \frac{r^2}{2} - \frac{r}{2}, \tag{1.12}$$

$$\mathrm{DOF}_{\mathrm{right}} = \sum_{i=1}^{r}(n - i) = nr - \frac{r^2}{2} - \frac{r}{2}, \tag{1.13}$$

$$\mathrm{DOF}_{\mathrm{values}} = r. \tag{1.14}$$

Consequently, when r is small, only a tiny fraction of the entries of $\boldsymbol{M}$ are necessary to fully determine the whole matrix.

Classical applications of low-rank matrix recovery are listed in E. J. Candes and Plan (2009). The most famous is probably *collaborative filtering*, especially the *Netflix Challenge*. In collaborative filtering, we are given an incomplete matrix containing ratings of some items—movies, in the case of Netflix—by some users, and we are asked to predict the rating a user would give to an item they have never encountered. The low-rank nature of the matrix comes from the idea that movies can be organized in a small number of "genres", which are rated similarly by the same kinds of user.

Notation. In the rest of this section, $\boldsymbol{M}$ and Ω are as defined above. $\mathcal{P}_\Omega : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ is the projection that replaces a matrix's entries in $\overline{\Omega}$ (the complement of Ω) with zero:

$$[\mathcal{P}_\Omega(\boldsymbol{X})]_{i,j} = \begin{cases} \boldsymbol{X}_{i,j} & \text{if } (i,j) \in \Omega, \\ 0 & \text{otherwise.} \end{cases} \tag{1.15}$$

$\mathcal{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^{|\Omega|}$ is the sampling operator associated to Ω, which collects the observed entries Ω of a matrix. Note that $\mathcal{P}_\Omega = \mathcal{A}^*\mathcal{A}$. Also, $\boldsymbol{b} = \mathcal{A}(\boldsymbol{M})$ is the vector of observed entries of $\boldsymbol{M}$. $\|\cdot\|_F$ designates the Frobenius norm and $\|\cdot\|_*$ the nuclear norm (the sum of singular values).

1.4.1 Rank Minimization

At the basis of low-rank matrix completion is the problem of *rank minimization* (RM). Assuming there is a unique low-rank matrix $\boldsymbol{M}$ that fits the observations Ω, it can be recovered by solving the following RM program (E. J. Candes and Recht 2008):

$$\begin{aligned} \min_{\boldsymbol{X}} \quad & \operatorname{rank}(\boldsymbol{X}) \\ \text{subject to} \quad & \mathcal{P}_\Omega(\boldsymbol{X}) = \mathcal{P}_\Omega(\boldsymbol{M}), \end{aligned} \tag{1.16}$$

or, if the measurements are noisy (E. J. Candes and Plan 2009):

$$\text{subject to} \quad \|\mathcal{P}_\Omega(\boldsymbol{X} - \boldsymbol{M})\|_F^2 \le \epsilon, \tag{1.17}$$

for some tolerance $\epsilon > 0$.

Unfortunately, RM is known to be NP-Hard (Vandenberghe and Boyd 1996). Most of the recent developments in the field of matrix completion have been concerned with finding conditions on $\boldsymbol{M}$ and Ω, as well as techniques, which guarantee a recovery in polynomial time. These techniques usually tackle either the RM problem directly—those are typically based on *hard thresholding*—or its relaxation into *nuclear norm minimization* (NNM)—often with *soft thresholding*. NNM is formulated as (E. J. Candes and Recht 2008):

$$\begin{aligned} \min_{\boldsymbol{X}} \quad & \|\boldsymbol{X}\|_* \\ \text{subject to} \quad & \mathcal{P}_\Omega(\boldsymbol{X}) = \mathcal{P}_\Omega(\boldsymbol{M}). \end{aligned} \tag{1.18}$$

1.4.2 Iterative Singular Value Thresholding

Soft Thresholding

Fazel (2002) introduced the idea of using the nuclear norm as a convex relaxation of the rank in the RM problem, which spawned a series of studies on NNM. E. J. Candes and Recht (2008) showed that, under some conditions on the matrix $\boldsymbol{M}$ and the sampling Ω, NNM produces the same (exact) solution as RM. Those conditions were later improved in E. J. Candes and Plan (2009), while also providing similar guarantees for the case when $\boldsymbol{M}$ is low-rank corrupted by additive noise.

Initially, NNM was solved by recasting the problem as semidefinite programming. However, this method becomes computationally prohibitive as the size of $\boldsymbol{M}$ increases. Cai et al. (2008) proposed a specialized algorithm, inspired by similar developments in compressed sensing, for solving NNM, named *iterative singular value* ***soft*** *thresholding* (ISVST).

As its name implies, ISVST is an iterative scheme that relies on the singular value soft thresholding operation $\mathcal{S}_\tau(\cdot)$ with threshold τ, defined as:

$$\mathcal{S}_\tau(\boldsymbol{X}) = \boldsymbol{U}\mathcal{S}_\tau(\boldsymbol{\Sigma})\boldsymbol{V}^\top, \quad \mathcal{S}_\tau(\boldsymbol{\Sigma}) = \operatorname{diag}(\{\max(\boldsymbol{\Sigma}_{i,i} - \tau, 0)\}), \tag{1.19}$$

where $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ is the SVD of $\boldsymbol{X}$.

The kth iteration of ISVST with step size δ_k is described by:

$$\begin{cases} \boldsymbol{X}^k = \mathcal{S}_\tau(\boldsymbol{Y}^{k-1}) \\ \boldsymbol{Y}^k = \boldsymbol{Y}^{k-1} + \delta_k \mathcal{P}_\Omega(\boldsymbol{M} - \boldsymbol{X}^k), \end{cases} \tag{1.20}$$

with $\boldsymbol{Y}^0 = 0$. This can be seen as an alternate optimization, over $\boldsymbol{X}$ and $\boldsymbol{Y}$, of the Lagrangian of:

$$\begin{aligned} \min_{\boldsymbol{X}} \quad & \tau\|\boldsymbol{X}\|_* + \frac{1}{2}\|\boldsymbol{X}\|_F^2 \\ \text{subject to} \quad & \mathcal{P}_\Omega(\boldsymbol{X}) = \mathcal{P}_\Omega(\boldsymbol{M}), \end{aligned} \tag{1.21}$$

using $\boldsymbol{Y}$ as the Lagrange multiplier. Note that (1.21) is very similar to the NNM problem, and indeed they are shown to converge to the same solution as $\tau \to \infty$.

Conditions under which this algorithm reaches an optimum of the RM problem are described in E. J. Candes and Plan (2009). They apply to the NNM problem in general. These conditions are that Ω be uniform random, and that $\boldsymbol{M}$ have *high incoherence*. Informally, a matrix with high incoherence has its data distributed fairly across its entries, such that random sampling may gather enough information for a successful recovery. For example, a low-incoherence matrix could be a rank-1 matrix that vanishes outside a given row. In this case, one has to sample every entry in that specific row, lest the matrix cannot be fully recovered.

Hard Thresholding

If the rank r is known, the RM problem (1.17) can be recast as:

$$\begin{aligned} \min_{\boldsymbol{X}} \quad & \frac{1}{2}\|\mathcal{P}_\Omega(\boldsymbol{X}) - \mathcal{P}_\Omega(\boldsymbol{M})\|_F \\ \text{subject to} \quad & \operatorname{rank}(\boldsymbol{X}) \le r. \end{aligned} \tag{1.22}$$

Under this formulation, the problem stays NP-Hard because the low-rank search space is nonconvex. However, using truncated SVD, it is very easy to compute the best projection of a matrix onto a rank-r subspace—this is essentially what PCA does. This operation suggests the following *iterative singular value **hard** thresholding* (ISVHT) scheme, introduced by Meka et al. (2009):

$$\begin{cases} \boldsymbol{Y}^k = \boldsymbol{X}^{k-1} - \mu_k \mathcal{A}^*(\mathcal{A}(\boldsymbol{X}^{k-1}) - \boldsymbol{b}) \\ \boldsymbol{X}^k = \mathcal{H}_r(\boldsymbol{Y}^k), \end{cases} \tag{1.23}$$

with $\boldsymbol{X}^0 = \mathcal{P}_\Omega(\boldsymbol{M})$, and where μ_k is a step size and $\mathcal{H}_r(\cdot)$ is the *hard thresholding* operator, which projects a matrix onto its r first singular vectors. The first part performs a gradient descent on the objective function, while the second enforces the low-rank constraint. This is repeated until convergence.

In a way, this algorithm can be seen as a generalization of PCA for matrices with missing entries. When $\overline{\Omega} = \emptyset$, $\boldsymbol{X}^0 = \boldsymbol{M}$, the gradient is zero and the method converges in a single step, reducing to PCA.

While ISVHT lacks the theoretical guarantees offered by NNM, Meka et al. (2009) have shown empirically that, under the same hypotheses, it performs just as well as its soft thresholding counterpart. Moreover, ISVHT tends to be more robust to outliers and noise because it minimizes the error between $\mathcal{A}(\boldsymbol{X})$ and $\boldsymbol{b}$.

1.4.3 Low-Rank Matrix Factorization

All the methods presented so far rely on SVD, either to minimize the nuclear norm or to enforce a low-rank constraint. Unfortunately, SVD is very expensive for large, dense matrices. Hence, when $|\Omega| \approx mn$, avoiding that costly operation can result in a great performance boost.

A way to do that is to replace in problem 1.22 the matrix $\boldsymbol{X}$ with two matrices $\boldsymbol{W} \in \mathbb{R}^{m \times r}$ and $\boldsymbol{H} \in \mathbb{R}^{n \times r}$ such that $\boldsymbol{X} = \boldsymbol{W}\boldsymbol{H}^\intercal$. In this form, the rank constraint is implicitly enforced and the need for SVD vanishes. This gives the *(sparse) matrix factorization* (MF) problem (see *e.g.* Lu and Yang 2015):

$$\min_{\boldsymbol{W},\boldsymbol{H}} \frac{1}{2}\|\mathcal{P}_\Omega(\boldsymbol{M} - \boldsymbol{W}\boldsymbol{H}^\intercal)\|_F^2 + \frac{\lambda_w}{2}\|\boldsymbol{W}\|_F^2 + \frac{\lambda_h}{2}\|\boldsymbol{H}\|_F^2. \tag{1.24}$$

Note that the rank constraint has disappeared, and that two regularization terms were added to the objective function.

LMaFit

Wen et al. (2012) suggested to solve a slightly altered problem, without the regularizers, and introducing the splitting variable $\boldsymbol{Z}$:

$$\begin{aligned} \min_{\boldsymbol{W},\boldsymbol{H},\boldsymbol{Z}} \quad & \frac{1}{2}\|\boldsymbol{W}\boldsymbol{H}^\intercal - \boldsymbol{Z}\|_F^2 \\ \text{subject to} \quad & \mathcal{P}_\Omega(\boldsymbol{Z}) = \mathcal{P}_\Omega(\boldsymbol{M}), \end{aligned} \tag{1.25}$$

A simple way to solve (1.25) is to set the differential of the Lagrangian to zero. This yields four systems of equations (one per variable $\boldsymbol{W}, \boldsymbol{H}, \boldsymbol{Z}$ and one for the Lagrange multiplier $\boldsymbol{\Lambda}$). These systems can each be solved with Gauss-Seidel elimination, giving an alternate optimization scheme. To accelerate convergence, Wen et al. (2012) used *successive over-relaxation* (SOR) instead of Gauss-Seidel (for more on SOR, see *e.g.* Barrett et al. 1994). This method is called *LMaFit*.

Matrix Factorization with Graph Information

In Rao et al. (2015), it was proposed to extend (1.24) to include side information in the form of a graph on the rows of $\boldsymbol{W}$ and $\boldsymbol{H}$. Each row of $\boldsymbol{W}$ corresponds to a vertex in a weighted graph whose weights encode similarities between the vertices. With $\boldsymbol{E}^w$ the adjacency matrix of this graph, the following term

$$\frac{1}{2}\sum_{i,j} \boldsymbol{E}^w_{i,j}\|\boldsymbol{w}_i - \boldsymbol{w}_j\|_2^2 = \mathrm{tr}(\boldsymbol{W}^\top \mathbf{Lap}(\boldsymbol{E}^w)\boldsymbol{W}) \tag{1.26}$$

enforces, when minimized, a (Euclidean) similarity between the rows $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$ of $\boldsymbol{W}$ whenever $\boldsymbol{E}^w_{i,j}$ is large. $\mathbf{Lap}(\boldsymbol{E}^w) = \boldsymbol{D}^w - \boldsymbol{E}^w$ is the graph Laplacian of $\boldsymbol{E}^w$, where $\boldsymbol{D}^w$ is the diagonal degree matrix, with $\boldsymbol{D}^w_{i,i} = \sum_{j\neq i} \boldsymbol{E}^w_{i,j}$.

With this additional term, (1.24) becomes:

$$\begin{aligned}\min_{\boldsymbol{W},\boldsymbol{H}} \frac{1}{2}\|\mathcal{P}_\Omega(\boldsymbol{M} - \boldsymbol{W}\boldsymbol{H}^\top)\|_F^2 + \frac{\lambda_L}{2}\Big[\mathrm{tr}(\boldsymbol{W}^\top \mathbf{Lap}(\boldsymbol{E}^w)\boldsymbol{W}) + \mathrm{tr}(\boldsymbol{H}^\top \mathbf{Lap}(\boldsymbol{E}^h)\boldsymbol{H})\Big] \\ + \frac{\lambda_w}{2}\|\boldsymbol{W}\|_F^2 + \frac{\lambda_h}{2}\|\boldsymbol{H}\|_F^2,\end{aligned} \tag{1.27}$$

which is written more compactly as:

$$\min_{\boldsymbol{W},\boldsymbol{H}} \frac{1}{2}\|\mathcal{P}_\Omega(\boldsymbol{M} - \boldsymbol{W}\boldsymbol{H}^\top)\|_F^2 + \frac{1}{2}\Big[\mathrm{tr}(\boldsymbol{W}^\top \boldsymbol{L}_w \boldsymbol{W}) + \mathrm{tr}(\boldsymbol{H}^\top \boldsymbol{L}_h \boldsymbol{H})\Big], \tag{1.28}$$

with $\boldsymbol{L}_w = \lambda_L \mathbf{Lap}(\boldsymbol{E}^w) + \lambda_w \boldsymbol{I}_m$, and similarily for $\boldsymbol{L}_h$. Because it uses graph Laplacians as regularizers, the method derived from (1.28) is appropriately called *graph-regularized matrix factorization* (GRMF).

While this problem has a closed-form solution, the computational load quickly becomes prohibitive for larger matrices. Instead, Rao et al. (2015) suggest an efficient implementation by applying the conjugate gradient method alternatively to $\boldsymbol{W}$ and $\boldsymbol{H}$—for more information, see the details in the paper.

1.4.4 Compressed Sensing

Many of the recent advances in matrix completion actually originate from, and take place in the context of, compressed sensing (E. J. Candes and Recht 2008). In order to be thorough, but without going into too much details, we present a short introduction to compressed sensing for data recovery.

Signal theory explains that, when measuring a signal, perfect recovery is assured if the Nyquist-Shannon sampling theorem is respected. With clever priors on the data, compressed sensing manages to reach full recovery with fewer samples (E. Candes et al. 2005).

Compressed sensing attempts to solve the underdetermined linear system:

$$\boldsymbol{y} = \boldsymbol{D}\boldsymbol{x} \tag{1.29}$$

where $\boldsymbol{y}$ has infinitely many representations as a vector $\boldsymbol{x}$ multiplied by the basis $\boldsymbol{D}$. Matrix completion imposed a low-rank prior in order to reduce the space of solutions. Compressed sensing does something similar with a sparsity prior on $\boldsymbol{x}$, promoting solutions with small ℓ_0 norm.

The concepts behind compressed sensing have a plethora of applications in signal processing. A common one is denoising, in which the signal $\boldsymbol{y}$ is corrupted with an additive noise. This signal is assumed to have a sparse representation $\boldsymbol{x}$ in the basis $\boldsymbol{D}$, while the noise does not. Denoising is achieved by finding a sparse $\boldsymbol{x}^*$ such that $\boldsymbol{y} \approx \boldsymbol{D}\boldsymbol{x}^*$. Then $\boldsymbol{y}^* = \boldsymbol{D}\boldsymbol{x}^*$ is the denoised signal.

Those techniques can also be applied to the problem of matrix recovery. Let $\boldsymbol{y}$ be the vectorized matrix $\boldsymbol{M}$ that, once again, has a sparse representation in $\boldsymbol{D}$. We can find $\boldsymbol{x}^*$ by solving the problem (1.29) restricted to the observed entries Ω, perhaps up to a tolerence if there is noise too, and $\boldsymbol{y}^* = \boldsymbol{D}\boldsymbol{x}^*$ is the recovered matrix.

The quality of this recovery greatly depends on the basis $\boldsymbol{D}$. One possible choice is to use a generic basis, *e.g.* discrete cosine transform or wavelets. Alternatively, good results can be obtained by constructing a dictionary from many reference signals.

Iterative singular value soft/hard thresholding are techniques that, originally, were created for compressed sensing. *Iterative hard thresholding* (IHT) is the equivalent of ISVHT (Blumensath and Davies 2009). It applies iterative entry-wise thresholding, instead of singular value thresholding, to solve the problem:

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{D}\boldsymbol{x}\|_2^2 \\ \text{subject to} \quad & \|\boldsymbol{x}\|_0 \leq s. \end{aligned} \tag{1.30}$$

Chapter 2

Tools and Experimental Data

In this section, we describe the tools, the data and the data augmentation techniques used in assessing the performance of our method. We begin by introducing the VIP toolbox, which was used throughout the writing of this thesis for pre- and post-processing tasks. We continue with a short description of the VLT/NaCo β Pictoris data cube. Finally, we explain fake companion injection, a common method for data augmentation.

2.1 The VIP Toolbox

The *Vortex Image Processing* (VIP) library is a `python` package for high-contrast imaging (Gonzalez, Wertz, et al. 2017). It includes a wide variety of tools for pre-processing, PSF estimation, and detection on ADI cubes.

ADI-based techniques depend heavily on the frames being aligned and having their star in the center. VIP comes with tools for centering and aligning that work by fitting a 2D Gaussian distribution to the data. It can also detect and correct or remove statistically bad pixels and bad frames.

Many state-of-the-art techniques for PSF estimation are included in the package, most notably PCA, LLSG and ANDROMEDA, which were covered in section 1.2. VIP also comes with detection tools for generating an SNR_t map and locating λ/D-wide patches on a frame. Finally, the library includes tools for fake companion injection, which are covered later in this chapter.

2.2 The ADI Dataset

The dataset we used for performance assessment in this thesis is an observation of the β Pictoris system in the infra-red L' band, made by the VLT/NACO telescope. It was described in details by Absil et al. (2013). The cube is composed of 612 frames captured over a period of 204 min, with parallactic angles ranging from $-15°$ to $68°$.

An additional observation of the system was performed without the coronagraph, in order to aquire the reference PSF of the telescope. This PSF describes the response of the telescope to an off-axis point source. It can be used as a model for the exoplanets, for instance in ANDROMEDA, in the matrix completion-based technique we introduce in this document, and in the fake companion injection technique described below. The reference PSF is also used to estimate the FWHM of Airy patterns.

We applied pre-processing to the cube with the VIP toolbox. The frames were recentered by fitting a Gaussian to the observations. They were also cropped from 300-by-300 pixels to 50-by-50, in order to reduce the computation load.

2.3 Fake Companion Injection

Direct imaging data is relatively scarce, making it hard to assess the performance of detection techniques and test their limits. By definition, it's only possible to test whether a method works if all the companions in a cube are known beforehand. This makes it hard to test whether a new technique can detect objects in conditions (intensity, radial distance) worse than what the state of the art can do. The small sample size also implies that one cannot achieve statistical significance in one's results. As a workaround, it has become common to work on artificial data.

Fake companion injection consists in taking an existing ADI cube and adding an artificial expolanet in it along a given trajectory. To inject a fake companion of intensity a in a cube $\boldsymbol{C}$ along trajectory g, a new cube $\boldsymbol{C'}$ is created in which the t-th frame is:

$$\boldsymbol{C'}(t) = \boldsymbol{C}(t) + a\boldsymbol{P}_g(t) \tag{2.1}$$

where $\boldsymbol{P}_g(t)$ is the telescope's impulse response $\boldsymbol{P}$, translated onto the t-th location in the trajectory g. Here, $\boldsymbol{P}$ is used as a model for the planetary signature.

This technique is very useful to study the performance of a detection method in different situations. For instance, to see the impact of radial distance on the method, one would create many cubes containing a single companion at various radii. The detection technique is then run on each cube and its results can be compared. This idea is explored more thoroughly in chapter 4.

Note that in order to produce a cube where only a single exoplanet appears at a chosen location, it is necessary to start with an "empty" cube containing no companion. This is achieved using two tricks. The first, *negative fake companion injection*, consists in injecting companions with negative flux into the cube at the location of the known planets. This suppresses most of the planetary signal. Then, in order to remove any lingering signal correlated along the trajectories, the cube is made to rotate in the opposite direction at the same rate. This preserves the star and noise distribution, but it makes sure that any residual planetary signal does not align when the cube is de-rotated.

Chapter 3

Proposed Solution

In this chapter, we outline the method developed in the context of this thesis. First, we explain our process for generating a model PSF from an ADI cube. Then, we describe the way we compute the flux and detection map in the context of our method. We finish with a discussion of the various matrix completion techniques presented in section 1.4 in the context of our method.

3.1 Description of Our Method

A common problem in techniques such as PCA and LLSG is companion absorption in the model PSF. Part of the flux emitted by a companion is included in the model and ultimately subtracted from the cube. This leads to the flux being underestimated, which might affect detectability.

To address this problem, we propose a solution in which a different model PSF is estimated for each potential trajectory. Fig. 3.1 illustrates this new technique. For a given trajectory g, a disk of diameter λ/D is removed along g on each frame of the ADI cube (top left). These entries are then filled by a chosen matrix completion technique, creating our model PSF (top right). Finally, the flux is estimated with a maximum likelihood approach (bottom) and a STIM score is computed along g. By repeating this process for every trajectory of interest in the cube, we create a flux map (*i.e.* processed frame) and a STIM detection map.

Matrix completion. Let $\boldsymbol{C}$ be a T-by-N-by-N ADI cube in which we want to find potential companions, and $\boldsymbol{P}$ be the reference PSF of the telescope. Let $\mathcal{M}(\boldsymbol{X}, \Omega; \theta) : \mathbb{R}^{T\times N^2} \times 2^{\mathbb{N}^2} \to \mathbb{R}^{T\times N^2}$ be a chosen matrix completion method, with $\boldsymbol{X}$ the input matrix, Ω the set of observed indices (*i.e.* the ones that weren't removed), and θ representing possible parameters for $\mathcal{M}$. Note that $2^{\mathbb{N}^2}$ is the power set of $\mathbb{N}^2$.

Let $g \in \mathbb{N}^{T\times 2}$ be the trajectory of interest, such that $\boldsymbol{g}_t \in \mathbb{N}^2$ is the trajectory's location on the t-th frame. In a first step, we construct the set of observed indices Ω_g for the trajectory g. Ω_g is a list of triples $(t, \boldsymbol{r})$, where t is a frame index and $\boldsymbol{r}$ is a pixel location. Formally, it is defined as:

$$\Omega_g = \left\{ (t, \boldsymbol{r}) \in \{0, \ldots, T-1\} \times \{0, \ldots, N-1\}^2 \;\middle|\; \|\boldsymbol{r} - \boldsymbol{g}_t\|_2 > \frac{1}{2}\frac{\lambda}{D} \right\}. \tag{3.1}$$

In other words, Ω_g is the list of cube indices that are not within $\frac{1}{2}\lambda/D$ of the trajectory g.

Using $\mathcal{M}$ and appropriate parameters θ, the indices $\overline{\Omega}_g$ in $\boldsymbol{C}$ are filled to create the model PSF $\hat{\boldsymbol{C}}_g$. Since $\mathcal{M}$ requires a matrix as input, $\boldsymbol{C}$ is first reshaped into a T-by-N^2 array by collapsing its spatial dimensions, the same way it was done for PCA. Similarily, the indices Ω are first mapped onto matrix indices. The output of $\mathcal{M}$ is then expanded back into a cube.

Flux estimation. As explained above, we use a maximum likelihood approach to estimate the flux along g. This approach is very similar to the one used by ANDROMEDA (section 1.2.4).

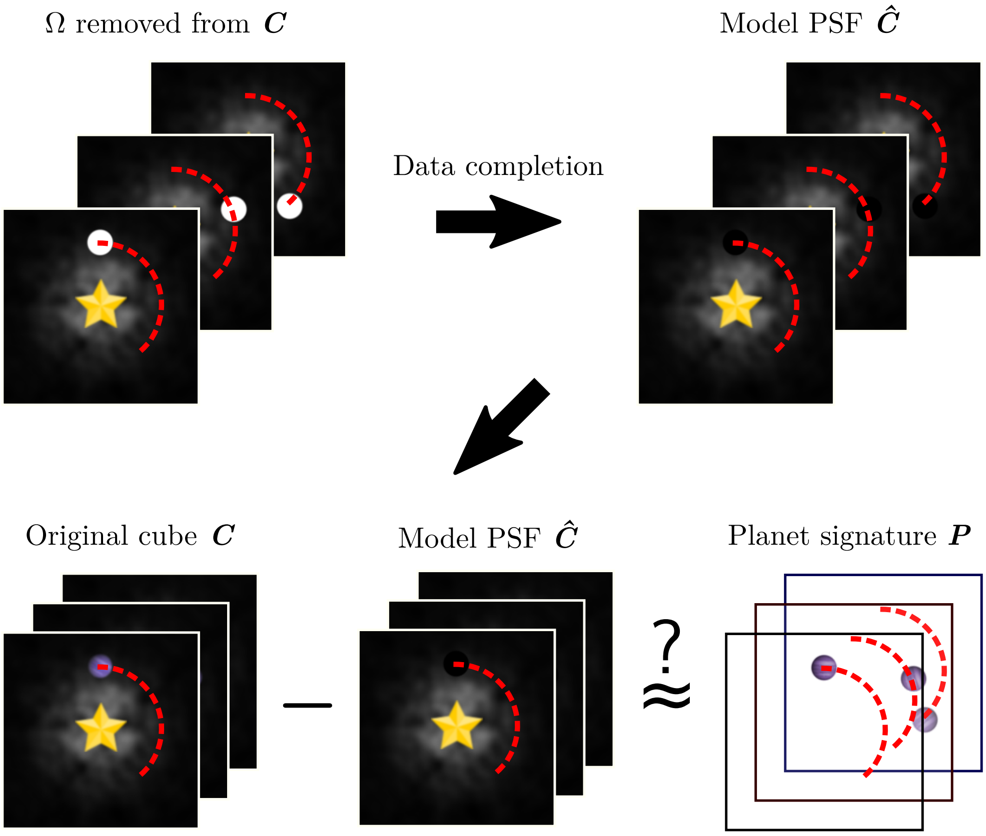

Figure 3.1: Illustration of the matrix completion-based PSF estimation method presented in this thesis. The dashed lines represent the trajectory g.

Assuming there is a planet on g, the residual cube $\boldsymbol{R}_g = \boldsymbol{C} - \hat{\boldsymbol{C}}_g$ is modeled as:

$$\boldsymbol{R}_g = a_g \boldsymbol{P}_g + \boldsymbol{N} \tag{3.2}$$

where a_g is the flux, $\boldsymbol{P}_g$ is the planet signature along g, and $\boldsymbol{N}$ is the residual noise. $\boldsymbol{P}_g$ is constructed by placing a copy of the reference PSF at location g_t on each frame t of a zero-filled cube.

If the noise $\boldsymbol{N}$ is i.i.d. Gaussian, we can use a maximum likelihood estimator to discover the value of a_g:

$$\log L_g^{\text{Gaussian}}(a) \propto -\frac{1}{2} \sum_{(t,\boldsymbol{r}) \in \overline{\Omega}_g} \frac{|\boldsymbol{R}_g(t, \boldsymbol{r}) - a\boldsymbol{P}_g(t, \boldsymbol{r})|^2}{\sigma^2_{\boldsymbol{R}_g}(\boldsymbol{r})}, \tag{3.3}$$

where $\sigma^2_{\boldsymbol{R}_g}$ is the empirical variance of the residual frames computed along the time dimension.

As explained in section 1.3.2, the Gaussian assumption for the residual noise is unrealistic. In particular, it dramatically underestimates the tails of the distribution. Pairet et al. (2018) discuss the nature of the residual noise and show that it tends follow a sub-exponential distribution, closer to a Laplacian than a normal. Knowing this, we choose to replace the Gaussian assumption in (3.3) with a Laplacian, which corresponds to substituting an ℓ_1 norm for the ℓ_2 norm:

$$\log L_g^{\text{Laplacian}}(a) \propto -\frac{1}{2} \sum_{(t,\boldsymbol{r}) \in \overline{\Omega}_g} \frac{|\boldsymbol{R}_g(t, \boldsymbol{r}) - a\boldsymbol{P}_g(t, \boldsymbol{r})|}{\sigma_{\boldsymbol{R}_g}(\boldsymbol{r})}, \tag{3.4}$$

which gives:

$$\begin{aligned} a_g &= \operatorname{argmax} \log L_g^{\text{Laplacian}}(a) \\ &= \operatorname{argmin} \sum_{(t,\boldsymbol{r}) \in \overline{\Omega}_g} \frac{|\boldsymbol{R}_g(t, \boldsymbol{r}) - a\boldsymbol{P}_g(t, \boldsymbol{r})|}{\sigma_{\boldsymbol{R}_g}(\boldsymbol{r})}. \end{aligned} \tag{3.5}$$

Unlike the Gaussian log-likelihood maximization, this expression does not have a closed form. However, it is still convex and can be solved efficiently, for instance with a quasi-Newton method.

The matrix completion and flux estimation processes are repeated for every trajectory of interest, and the fluxes are aggregated into a single image, the processed frame. To each pixel of this frame corresponds a trajectory in the cube (since trajectories become straight lines after derotation). Hence, the intensity of each pixel in the processed frame can be set to the estimated flux along the corresponding trajectory.

Detection. In addition to estimating the flux, we also output a detection value d_g for the trajectory g in order to construct a detection map. In section 1.3, two state-of-the-art detection maps we introduced: The SNR_t map and the STIM map. Because of its theoretical advantages and computational efficiency, we decided to use the latter. Here again, those results are aggregated into a single frame, which is our detection map.

An example of processed frames and STIM detection maps is shown on Fig. 3.2. The first couple of images were generated with PCA, while the rest are the product of matrix completion methods applied to PSF estimation, as described in this section. On all six images, the planet β Pictoris b appears clearly in the bottom right.

Additional notes. Since the whole matrix completion process must be repeated for each trajectory, speed constitutes the main drawback of our method. Even if it takes only a mere second to compute one model PSF $\hat{\boldsymbol{C}}_g$ and its associated flux and detection scores (a_g, d_g), on a 100-by-100 frame, the whole process will take up to three hours. On the bright side, since the computation for one trajectory is completely independent from that of the others, the technique is easily split into parallel tasks.

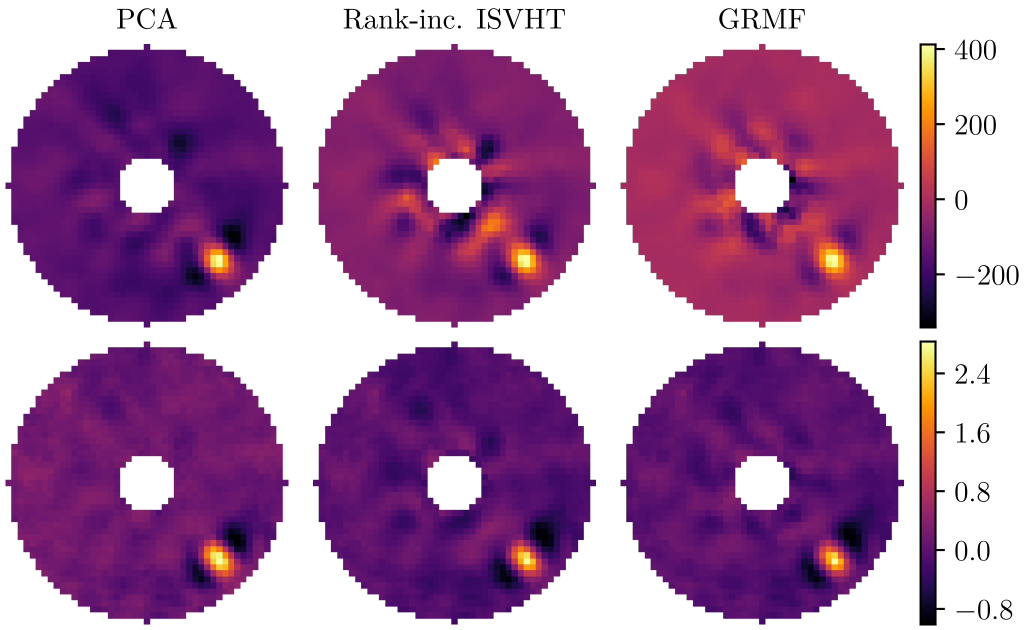

Figure 3.2: Processed frame (top) and STIM detection map (bottom) for various methods, computed on the β Pictoris dataset.

3.2 Matrix Completion Methods for PSF Estimation

We now return to some of the matrix completion methods introduced in section 1.4, and describe how we adapted them to the PSF estimation method presented above.

3.2.1 Iterative Singular Value Hard Thresholding

Because ADI data is only approximately low-rank, the matrix completion methods we employ must be robust to the presence of (non-Gaussian) residual noise. Between the soft and hard singular value thresholding techniques, the latter was shown to produce better results in the presence of noise, making it an ideal candidate for our purposes.

While both soft and hard thresholding come with remarkable recovery guarantees when the sampling is uniform, their performance can suffer in non-uniform settings such as ours. In particular, when the sampling operator has too small a rank, no matrix completion takes place. Unfortunately, this is the case for trajectories close to the star.

The closer a trajectory is to the center, the less it will change from frame to frame, because its "speed" becomes smaller (see Fig. 3.3). Mathematically, this means that trajectories have increasingly lower rank as they come closer to the star. Fig. 3.4 shows a linear relation between the two. In the case of PCA, this means that planets closer to the star are more absorbed into the first few principal components than those further away, making them harder to detect.

When applying ISVHT to a trajectory close to the star, a similar problem arises. As a reminder, ISVHT alternates between a projection onto the observed entries Ω and a thresholding of the singular values. The projection introduces holes in the model PSF, but when the trajectory's rank is too large, these holes cannot be represented by the model and they are filled during the thresholding step. However, when the trajectory and model have similar ranks, the holes tend to stay, as shown in Fig. 3.5.

To solve this problem we decided to adopt a rank-increasing strategy. ISVHT is first applied with a small rank (*e.g.* 1), which is then slowly increased everytime the relative error stalls for

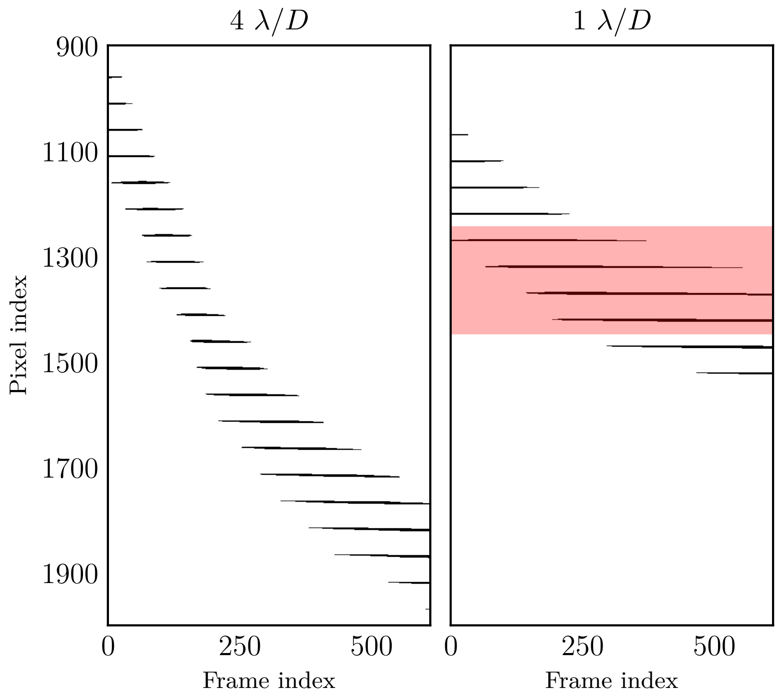

Figure 3.3: Pixels covered by a trajectory, displayed on a cube flattened to a matrix of pixels-by-frames, for two trajectories at radial distances $4\,\lambda/D$ (*left*) and $1\,\lambda/D$ (*right*). The shaded area highlights pixels which appear in less than 50 % of the frames.

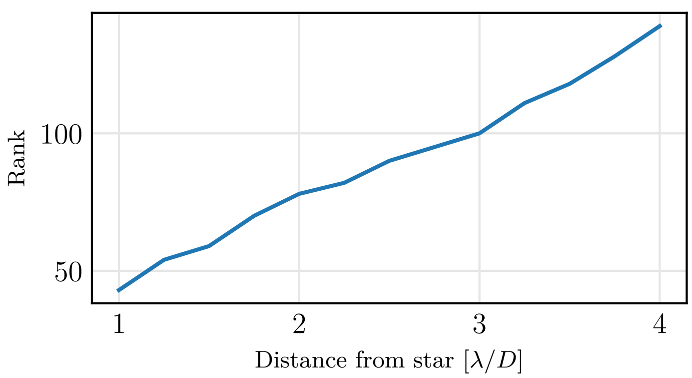

Figure 3.4: Rank of a trajectory as a function of the distance from the star.

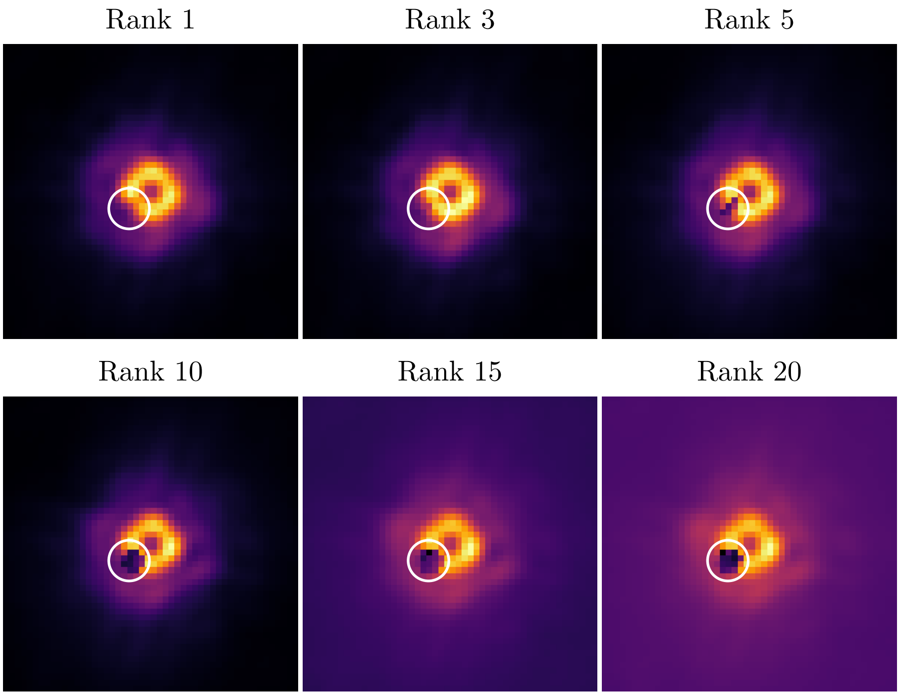

Figure 3.5: A single frame of the model PSF $\hat{\boldsymbol{C}}$ produced by ISVHT, for a trajectory at $1\,\lambda/D$ from the star, for various ranks. The area where pixels were imputed is circled. At higher ranks, the trajectory is included in the model.

too long, until the desired rank is reached. We call this method *rank-increasing ISVHT*. This strategy allows for a smarter search through the non-convex solution space, producing better models when close to the star. Fig. 3.6 shows the same images as Fig. 3.5, but produced with this method. We can see that the model PSF is no longer overfitted to the trajectory.

3.2.2 LMaFit

Similarily to ISVHT, the main parameter for LMaFit is the desired rank of the model. It is possible to give a fixed rank to the method, but Wen et al. (2012) suggest two possible policies for automatic rank selection. The first is a decreasing-rank strategy, and the second is an increasing-rank one. For the same reasons as ISVHT, we decided to work with an increasing rank.

3.2.3 Graph-Regularized Matrix Factorization

GRMF gives us the choice of a graph and weights for both dimensions of the matrix—in our case, the spatial and temporal dimensions. In the first graph, vertices represent pixels of the cube, and the weights enforce similarity between two pixels. In the second graph, vertices represent whole frames, and the weights enforce similarity between two frames. The way those graphs are constructed allow to impose many different constraints on the resulting model.

A possible approach is to enforce a *non-local* structure in space. If the frames contain patterns that repeat in several places, it might be interesting to build a non-local graph from the data, with connexions based on similarities rather than distances. In that vein, Dong et al. (2018) propose a graph learning algorithm that builds an adjacency matrix encoding similarities between the rows or columns of a matrix. With images, it is common to consider similarities between

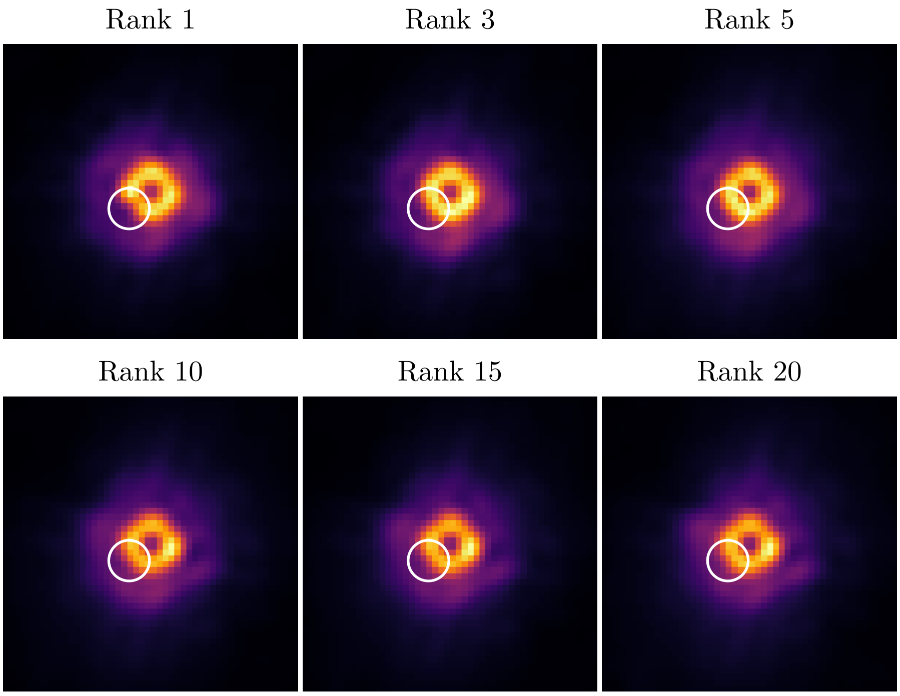

Figure 3.6: The same frames of the model PSF $\hat{\boldsymbol{C}}$ as Fig. 3.5, for the rank-increasing ISVHT method. The area where pixels were imputed is circled.

neighborhoods rather than individual pixels, as is done in *non-local means denoising* (Buades et al. 2005).

The approach we considered was to use the graphs to enforce similarity based on the spatial and temporal correlations. Spatially, the frames are the result of a convolution with the reference PSF of the telescope, which is known. Consequently, we expect the spatial correlation to be accurately described by that PSF. We used it to construct a graph in the following way. Let $\boldsymbol{P}$ be the reference PSF centered in $(0,0)$. The weight $w_{\boldsymbol{r},\boldsymbol{r}'}$ between pixels $\boldsymbol{r}$ and $\boldsymbol{r}'$ is given by:

$$w_{\boldsymbol{r},\boldsymbol{r}'} = \boldsymbol{P}(\boldsymbol{r}' - \boldsymbol{r}). \tag{3.6}$$

In other words, the weight between two pixels is obtained by placing the reference PSF on the first pixel, and looking at the value it takes on the second.

In practice, since the weights have to be symmetrical, the average of $\boldsymbol{P}(\boldsymbol{r}' - \boldsymbol{r})$ and $\boldsymbol{P}(\boldsymbol{r} - \boldsymbol{r}')$ is used. The reference PSF is also truncated beyond a given radius, to avoid over-constraining the model.

Temporally, since we are not given an estimate of the correlation, we need to construct one ourselves. We do this by considering each individual pixel $\boldsymbol{r}$ as a stationary time series $\boldsymbol{x}_{\boldsymbol{r}}(t) = \boldsymbol{C}(t, \boldsymbol{r})$, taking as many values as there are frames. We then compute the autocorrelation of those series, giving us the vectors $\boldsymbol{c}_{\boldsymbol{r}}$, where $\boldsymbol{c}_{\boldsymbol{r}}(u)$ is the correlation between $\boldsymbol{x}_{\boldsymbol{r}}(t)$ and $\boldsymbol{x}_{\boldsymbol{r}}(t \pm u)$ for all t. Finally, those vectors are averaged together entry-by-entry to give a single vector $\boldsymbol{c}$, representing the autocorrelation between the frames of the cube.

The graph is constructed similarily to the spatial dimension. The weight $w_{t,t'}$ between two frames with indices t and t' is given by:

$$w_{t,t'} = \boldsymbol{c}(t' - t). \tag{3.7}$$

In real ADI datasets, it is not uncommon for the angular speed to vary in time, making the angular step non-constant. This is why, for instance, the patterns on Fig. 3.3 do not lie on a straight line. Unfortunately, this means that the way we compute the temporal correlations $\boldsymbol{c}_{\boldsymbol{r}}$ is wrong, because it assumes a constant time step. The solution we employ is to interpolate the cube with an angular step ϕ that stays the same throughout. The correlation $\boldsymbol{c}$ can then be computed on the interpolated cube and the graph is constructed in this way:

$$w_{t,t'} = \boldsymbol{c}\left(\frac{\theta(t') - \theta(t)}{\phi}\right), \tag{3.8}$$

where $\theta(t)$ is the parallactic angle of the t-th frame, and $\boldsymbol{c}$ is interpolated for non-integer indices. Just like the spatial correlation, $\boldsymbol{c}$ is actually first truncated beyond a certain angle.

3.3 Combined Landscape of the State of the Art

To conclude this chapter, we draw up a short summary of the matrix completion techniques presented in section 1.4, now in light of PDF estimation. Table 3.1 shows a classification of the state of the art in ADI post-processing, together with matrix completion. Combining the two in this way makes sense because a completion method can be adapted for PSF estimation using the process described in the previous sections of this chapter.

The techniques are organized according to the usage they make of temporal and spatial data to create a model. When constructing the frame t (resp. pixel $\boldsymbol{r}$) of a model PSF, *global* means that all other frames (resp. pixels) are weighted equally. *Local* indicates that frames close in time (resp. pixels close in space) are weighted more heavily. *Non-local* means that some non-local weighting is used, *e.g.* based on similarities. c-ADI is classified in both temporal columns because the first step (global median) is global, while the second (4-frame median) is local.

		Temporal	
		Global	Local
Spatial	Global	PCA LLSG **ISVHT**/ISVST **LMaFit**	Wavelet-based denoising **GRMF (temporal correlation graph)**
	Local	**GRMF (spatial correlation graph)**	**GRMF (correlation graphs)**
	Non-Local	GRMF (non-local spatial graph)	GRMF (non-local spatial graph and temporal correlation graph)
	Not used	c-ADI (global median)	c-ADI (4-frames median) ANDROMEDA

Table 3.1: Summary of the state of the art in ADI post-processing, as well as matrix completion in the context of PSF estimation. Names in bold highlight the completion methods we explored. The techniques are organized according to their temporal global/locality, and spatial global/local/non-locality.

Chapter 4

Performance Assessment

4.1 Signal Detection Theory

We use the framework of *signal detection theory* (SDT) to assess the quality of detection techniques. The aim of SDT is to measure one's ability to differentiate a signal containing information from one containing pure noise.

Let T be a test statistic for a random signal X. For example, X might be a trajectory in a residual cube, and T its STIM score. Like in section 1.3, there are two opposing hypotheses: H_0 meaning X is only noise, and H_1 meaning X contains a companion.

Let τ be a threshold for the test T, such that H_0 is rejected when $T \geq \tau$. There are four possible outcomes, based on whether or not H_0 is true, and whether or not it was rejected. They are summarized in the table below.

	Companion present (H_1)	Companion absent (H_0)
Something detected	True positive (TP)	False positive (FP)
Nothing detected	False negative (FN)	True negative (TN)

The quality of a detection technique is measured in term of two quantities: The *true positive rate* (TPR) and the *false positive rate* (FPR), defined as:

$$\mathrm{TPR}(\tau) = \Pr(T \geq \tau | H_1) = \int_{\tau}^{+\infty} p(t|H_1)\,\mathrm{d}t \tag{4.1}$$

and

$$\mathrm{FPR}(\tau) = \Pr(T \geq \tau | H_0) = \int_{\tau}^{+\infty} p(t|H_0)\,\mathrm{d}t, \tag{4.2}$$

where $p(t|H_i)$ is the probability density function of T under the hypothesis H_i. The former measures one's ability to correctly detect all the existing companions, while the latter measures the tendency to mistake noise for a planetary signal. The FPR corresponds directly to the confidence level (CL) discussed in section 1.3, with the relation $\mathrm{FPR} = 1 - \mathrm{CL}$.

One would like to achieve as high a TPR and as low an FPR as possible, but in practice the two are linked and depend on the value of τ. Increasing the threshold leads to a smaller FPR, but it also decreases the TPR. However, not all detection techniques are equal, and some will produce better TPR for the same FPR. We can visualize the quality of a technique by plotting its *receiver operating characteristic* (ROC) curve, which displays the TPR as a function of the FPR. An example of ROC curves is displayed in Fig. 4.1.

When a technique is not better than randomly choosing H_0 or H_1, the ROC curve will trace a straight diagonal line from $(0, 0)$ to $(1, 1)$. In the case of a perfectly separable signal, the curve will go straight up from $(0, 0)$ to $(0, 1)$, meaning that a 1.0 TPR can be achieved with no false positives. Integrating the *area under the ROC curve* (AUC) gives a metric of how good an ROC

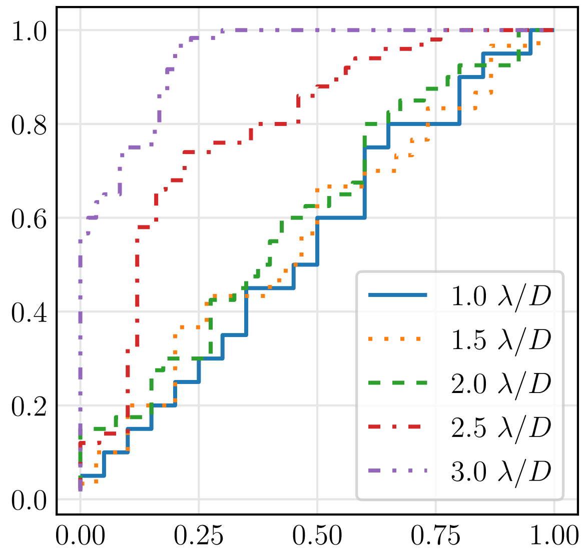

Figure 4.1: An example of classical ROC curves. The data which produced this was acquired with the process described in section 4.2, applied to the ISVHT method.

curve is. An AUC of 0.5 corresponds to a random classifier, while 1.0 corresponds to a perfectly separable signal.

In real settings, the PDFs $p(t|H_i)$ are not available, and the TPR and FPR must be measured empirically. To do so, we use a large labeled dataset containing many realizations of X under both hypotheses. The test T is computed on each sample, yielding a number of TP, FP, TN and FN as a function of the threshold. The TPR and FPR are then computed using the formulae:

$$\mathrm{TPR}(\tau) = \frac{\mathrm{TP}(\tau)}{\mathrm{TP}(\tau) + \mathrm{FN}(\tau)} \qquad \text{and} \qquad \mathrm{FPR}(\tau) = \frac{\mathrm{FP}(\tau)}{\mathrm{FP}(\tau) + \mathrm{TN}(\tau)}. \tag{4.3}$$

4.2 ROC Curves for Direct Imaging

As explained in section 2.3, because there aren't many labeled datasets available, quality assessment requires some kind of data augmentation which is achieve through fake companion injection. A labeled sample is created by taking a cube without companions and injecting one or more somewhere. This also creates an opportunity to study the performance of a detection technique as a function of the radial distance and intensity of the planet.

One problem with the typical definition of an ROC curve is that a full ADI cube containing n planets will yield n TP or FN, but many more FP or TN (in fact, as many as there are empty resolution elements). Counting every TN means that the FPR will always be very low, even when there are many false positives. An alternative could be to only count a single FP or TN per cube, but then it does not matter whether there is one false detection or a hundred.

TPR vs mean FP ROC curve. Gomez Gonzalez et al. (2017) proposed instead to compute an ROC curve that shows the TPR, not against the FPR, but against the average number of FP per cube. An example is shown on Fig. 4.2.

The complete scheme they proposed goes as follows. First, many datasets are created by injecting a planet of desired intensity at the chosen distance for various initial angles. Then, a detection map is computed from each dataset, and is thresholded with various values τ. Any pixel above τ in a λ/D region is considered to be a single detection. If a detection matches the

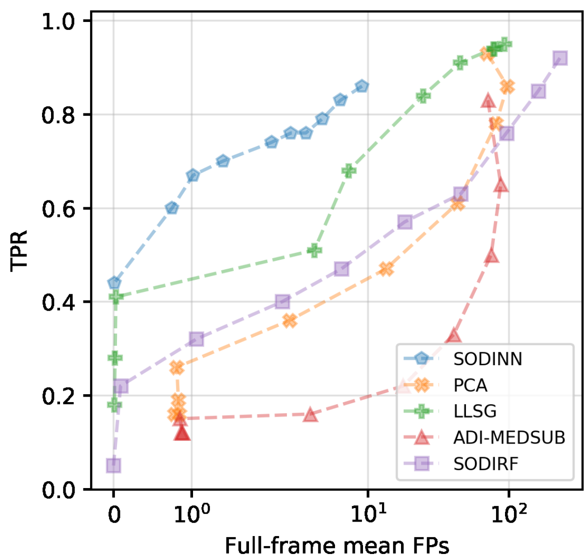

Figure 4.2: An example ROC curve for direct detection that uses TPR vs mean FP. Credit: Gomez Gonzalez et al. (2017).

injected fake companion, then it is counted as a TP. Any detection that does not is counted as an FP. If the fake companion is not matched, then a FN is counted as well.

ROC curves for our technique. While this process could be used to assess the quality of our method, because of the prohibitive time it takes to generate a single detection map, we could not handle the computation of hundreds of them (for each radius and intensity, and for each matrix completion technique considered). Instead, we propose the following scheme.

In our process, the TPR and the FPR are computed separately. First, we acquire the number of TP and FN in the following way. We generate a handful of cubes with a single injected planet at the radius and intensity of interest. For each of these cubes, rather than computing the full detection map, we only compute the model PSF and STIM score that corresponds to the planet's trajectory. A value above τ is counted as a TP and a value below is counted as a FN. This is done efficiently by calling our technique only once per cube.

The FP and TN are computed in a similar way, but on an empty cube. Several locations are picked uniformly at random at the radius of interest. For each location, a single model PSF and STIM score is computed. A value above τ is counted as an FP, and a value below is counted as a TN. Finally, the ROC curve is constructed in the traditional way, as the TPR versus FPR.

For this method to be accurate, two assumptions must be filled. The first is that the center of an injected exoplanet yields the highest value associated with this planet on a detection map. If this weren't the case, our method would underestimate the number of FN. This was verified empirically to be the case most of the time.

The second assumption is that the STIM score at an empty location is independent of the presence of a planet somewhere in the cube. This allows us to separate the computation of the TPR from that of the FPR. We have no formal theoretical result to back this up. However, this should have little effect on local methods because only close pixels have a great impact, and on global methods because the pixels containing a planet are "drowned" among the rest.

4.3 Quality of Flux Estimation

The method described above allows us to measure the quality of the detection. We might also be interested to know how well a given technique manages to estimate the astronomical flux of a companion. For instance, because part of the planetary signal is absorbed in the first few principal components, PCA tends to under-estimate the flux.

The way we measure the quality of a technique's flux estimation is with the *root mean square error* (RMSE) of relative flux. Given a set of n independent estimates $\hat{a}_i$ of the true flux a, the RMSE is defined as:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{\hat{a}_i - a}{a}\right)^2}. \tag{4.4}$$

The closer this value is to 0, the better a technique manages to estimate planetary flux.

The estimates $\hat{a}_i$ are obtained from empty cubes in which a planet of intensity a has been injected. In fact, they come as a byproduct of the process described in the previous section.

Chapter 5

Results and Discussion

We now use the framework presented in the previous chapter to assess the performance of the technique developed for this thesis.

5.1 Comparison Against the State of the Art

In this section, we evaluate the various matrix completion methods presented in section 3.2. Their performance are compared against the state of the art PCA technique, as well as what we call their "bypassed" equivalent. When a method is bypassed, it is run without removing the trajectory (*i.e.* setting $\overline{\Omega} = \emptyset$), therefore "bypassing" the matrix completion. Doing so, we verify whether any improvement is truly the result of the completion.

Table 5.1 lists those methods, along with the parameters we used. Note that, as shown in section 1.4.2, the bypassed equivalent of rank-increasing ISVHT is PCA itself.

It is important to state that PCA and bypassed methods were evaluated with the same performance assessment pipeline as the rest. Moreover, the flux estimation is done in the same way too (*i.e.* with a maximum likelihood estimator) rather than by averaging the residual cube. This is to assure that observed differences come from the techniques themselves, and not from the evaluation process.

5.1.1 Detection

Fig. 5.1 shows the AUC of each method as a function of the distance from the star (horizontally) and the flux of the injected planet (vertically). The brighter the color, the closer it is to a perfect classifier. The top row shows the bypassed methods, with PCA as our benchmark on the left. The bottom row shows the non-bypassed methods. Those AUC values were obtained as described in section 4.2.

The first thing to note is that all methods perform worse at smaller orbits. There are two main reasons that explain this. The first is that noise variance is much greater in the center, and a companion clearly visible in the outskirts becomes drowned in noise closer to the star. The second is that, at smaller distances, the planet overlaps more with itself between frames, effectively reducing the usability of the data.

For reasons unknown, LMaFit produces poor results when bypassed. Therefore, it will be ignored (only when bypassed) for the rest of this analysis.

Unfortunately, results for matrix-completion based methods do not match the state of the art. The performances are close to PCA, although slightly inferior. However, it is interesting to note that the bypassed version of GRMF gives results slightly above PCA. Hence, while the matrix completion is not conclusive here, the use of graph regularization for the construction of a model PSF might be promising.

Method	Parameter	Value
PCA	Rank	17
LMaFit (bypassed)	Init. rank	1
	Max. rank	17
	Rank increment	1
GRMF (bypassed)	Rank	20
	Graphs constructed as in section 3.2	
	Temporal corr. length	5°
	Spatial corr. radius	$2 \times \frac{1}{2}\lambda/D$
	$(\lambda_w, \lambda_h, \lambda_L)$	$(20, 160, 1)$
Rank-inc. ISVHT	Init. rank	1
	Max. rank	17
	Step size μ	1.0
	Increase rank when stalling for 5 iterations	
LMaFit	Same as bypassed	
GRMF	Same as bypassed	

Table 5.1: The methods and parameters used to assess the performance of our technique for PSF estimation with matrix completion. The first three methods are benchmarks without actual completion. Correlation length or radius corresponds to the truncation point for the correlation.

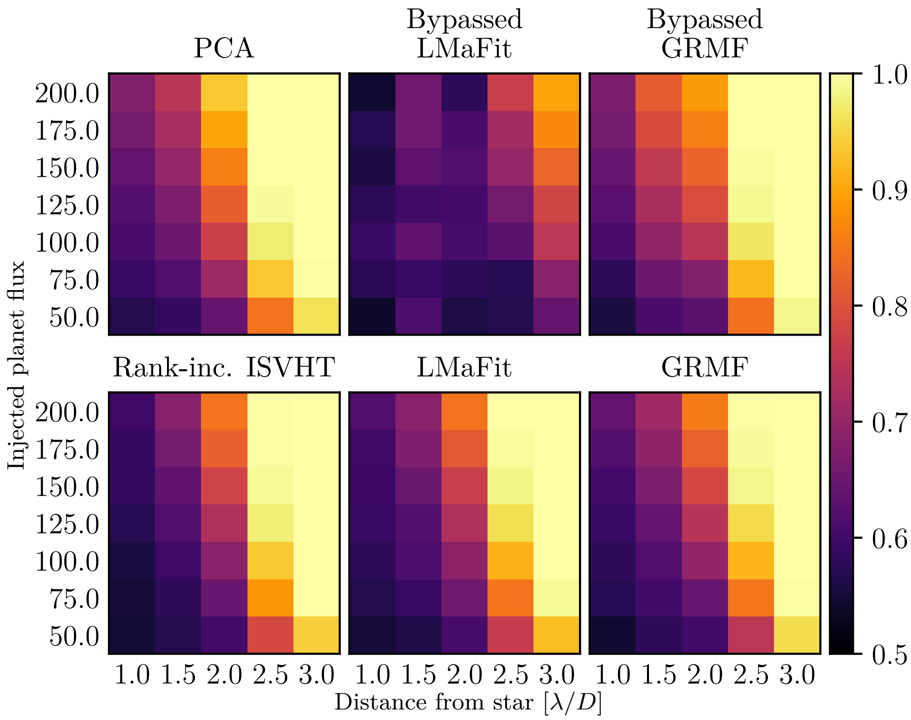

Figure 5.1: Area under the ROC curve for multiple methods, as a function of the injected flux intensity and distance from the star. Higher is better. The first row shows results without matrix completion ("bypassed"), while the second row shows the same methods with matrix completion. The bypassed equivalent of ISVHT is the classical PCA technique.

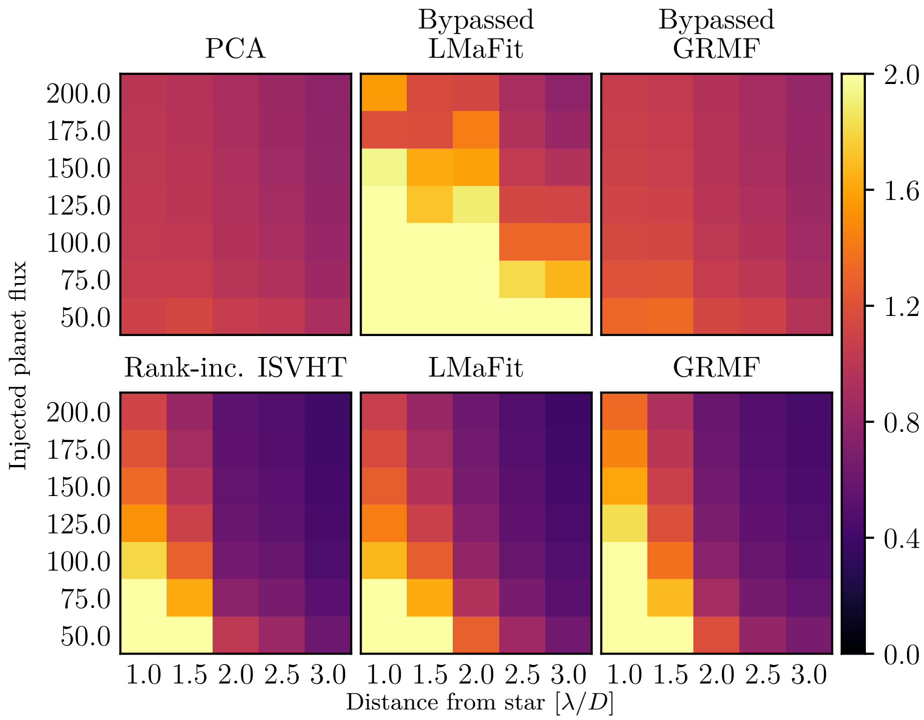

Figure 5.2: RMSE of relative flux for multiple methods, as a function of the injected flux intensity and distance from star. Smaller is better. Similarily to Fig. 5.1, the first row shows bypassed methods, while the second shows actual matrix completion results. To improve readability, large values were capped at 2.0.

5.1.2 Flux estimation

We now compare how well the various detection techniques manage to estimate planetary flux. As explained earlier, because part of the planetary signal is absorbed in the first few principal components, PCA tends to under-estimate the flux level of companions, especially closer to the star. Since matrix completion-based techniques remove the planetary signal before performing PSF estimation, this shouldn't be a problem, and we expect those methods to give better estimates of the flux.

Fig. 5.2 shows the RMSE of relative flux for the same methods and conditions as above. Similarly to the AUC plots, the main trend is that flux estimates are worse closer to the star. Once again, this is likely explained by the fact that the noise variance increases as we get closer to the center. Estimation also gets worse at smaller intensities, because the RMSE is computed on the estimated flux $\hat{a}$ normalized by the real flux a. Hence, when a decreases, the noise has more relative importance.

As expected, the techniques based on matrix completion perform on average better, although with greater variance, than those that use the full cube for PSF estimation. However, their performance tends to become worse as we get closer to the star. This is because matrix completion produces estimates with greater variance than PCA, especially at smaller radii. Fig. 5.3 highlights this phenomenon. It shows, for the same techniques as before and for various distances from the star, boxplots of the flux estimated on an empty cube. Ignoring LMaFit, bypassed methods are more or less on point and show little variance. In contrast, matrix completion-based estimations vary by a lot.

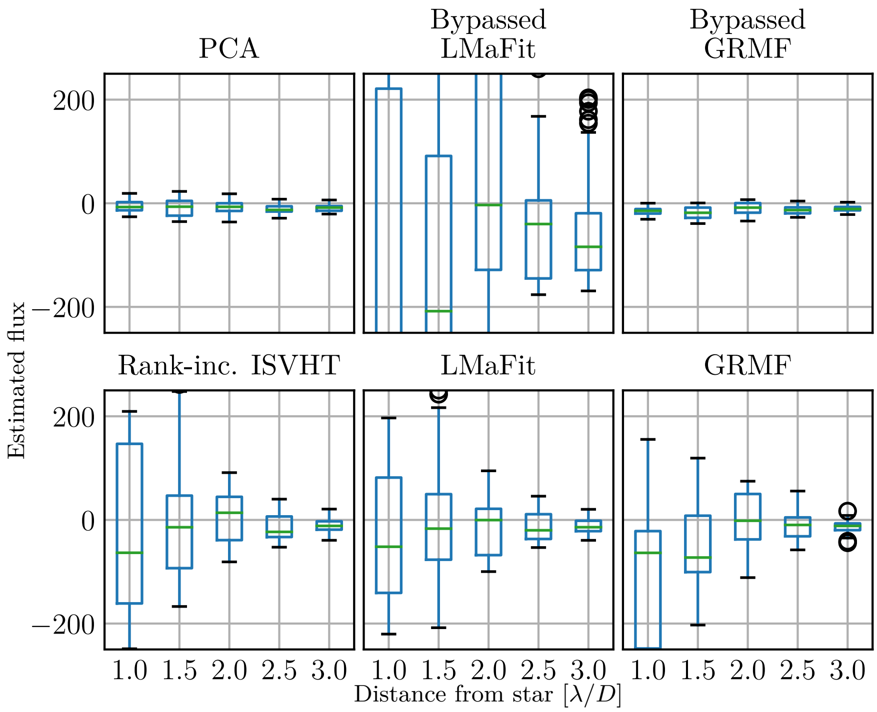

Figure 5.3: Flux estimated on an empty cube, as a function of the distance from the star, for various detection techniques.

5.1.3 Running Time

As explained earlier, because each trajectory requires a model PSF estimation, the technique we introduced is much slower than the state of the art. Hence, it makes little sense to compare them on that front. Instead, we evaluate the speed of matrix completion methods against one another.

Before we go further, it should be noted that the implementations we used of GRMF (and, to some extent, of LMaFit) does not have a good stopping criterion for our needs, and thus relies mostly on a maximum number of iterations. We fixed this number arbitrarily to a value that gives a reasonable trade-off between speed and accuracy, but it might be possible to achieve similar results in less time. Therefore, the speed analysis might be over-pessimistic.

Fig. 5.4 shows the average running time of the three completion methods considered in this work. Since the main parameter impacting speed is the rank, we fixed this value to 17 for all three methods. While LMaFit and ISVHT were roughly equivalent in terms of AUC and flux, the latter has a running time shorter by a couple of seconds. This is unexpected, as ISVHT uses singular value decomposition, a costly operation avoided by LMaFit. A possible expanation is that the method has trouble converging and often hits the maximum iteration limit we set. This is supported by the high variance observed it the method's running time.

5.2 Parameter Analysis

We now study the effect of parameters of the matrix completion methods. In what follows, for each parameter, we display a series of AUC heatmaps showing performances for different values of that parameter. Because the variations are often subtle, each heatmap is accompanied by a differential plot, showing the difference between the AUC heatmap and a reference. The reference is chosen among the heatmaps for the current parameter, and signaled with a † symbol.

Note that, since LMaFit produces results very similar to rank-increasing ISVHT, we decided

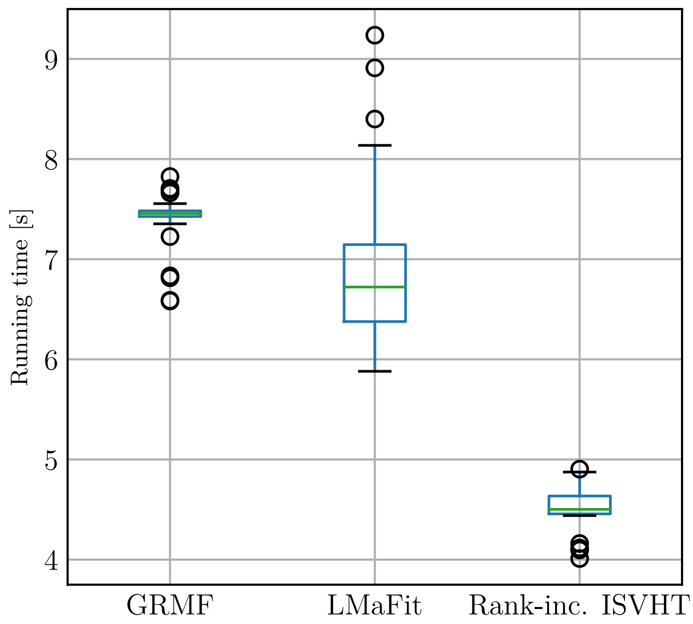

Figure 5.4: Average running time of matrix completion methods used for PSF estimation. Each method was run 100 times with a random distance and angle. The computation of the graph for GRMF was done beforehand. These results were obtained on a computer with a quad-core 2.70 GHz Intel® Core™ i7-7500U processor and 8 GB of RAM.

to perform parameter analysis on the latter only.

5.2.1 Rank-increasing ISVHT

Fig. 5.5 displays the performance results of rank-increasing ISVHT for various ranks. There seems to be a kind of "sweet spot" around rank 20 that gives optimal results. However, careful observation shows that at $1.5\,\lambda/D$, a rank of 40 is actually slightly better. Therefore, it might be interesting to split the cube into annuli and use a different rank for each.

5.2.2 GRMF

GRMF has many possible parameters to tune. We attempt to study the effect of some of them on the performances. Here too, the rank of the model is probably the most important parameter. We are also interested in the regularization factors λ_w and λ_h, for the temporal and spatial graphs respectively. Note that the effect of these factors depends on the scaling of the weights. We normalized the correlations $\boldsymbol{c}$ and $\boldsymbol{P}$ such that $\boldsymbol{P}(\boldsymbol{0}) = 1$ and $\boldsymbol{c}(0) = 1$. Finally, we also study the effect of different maximum lengths and radii for those correlation kernels. The length of $\boldsymbol{c}$ is expressed in degrees, while the radius of $\boldsymbol{P}$ is given as a multiple of $\frac{1}{2}\lambda/D$.

Figs. 5.6 through 5.10 show the performance results of parameter analysis for GRMF. Fig. 5.6 shows the analysis of the rank, Fig. 5.7 of the temporal regularization factor λ_w, Fig. 5.8 of the spatial regularization factor λ_h, Fig. 5.9 of the temporal correlation length, and Fig. 5.10 of the spatial correlation radius.

Like with ISVHT, good results are achieved with a rank of 20. Interestingly, while ISVHT performs on average worse with smaller and larger values, GRMF does almost as good, if not better. It seems the graph prior manages to make up for the loss brought by a sub-optimal rank.

Analysis of the temporal regularizer shows the noticeable impact of introducing a temporal correlation graph in the low-rank model. When the regularization factor becomes large (around

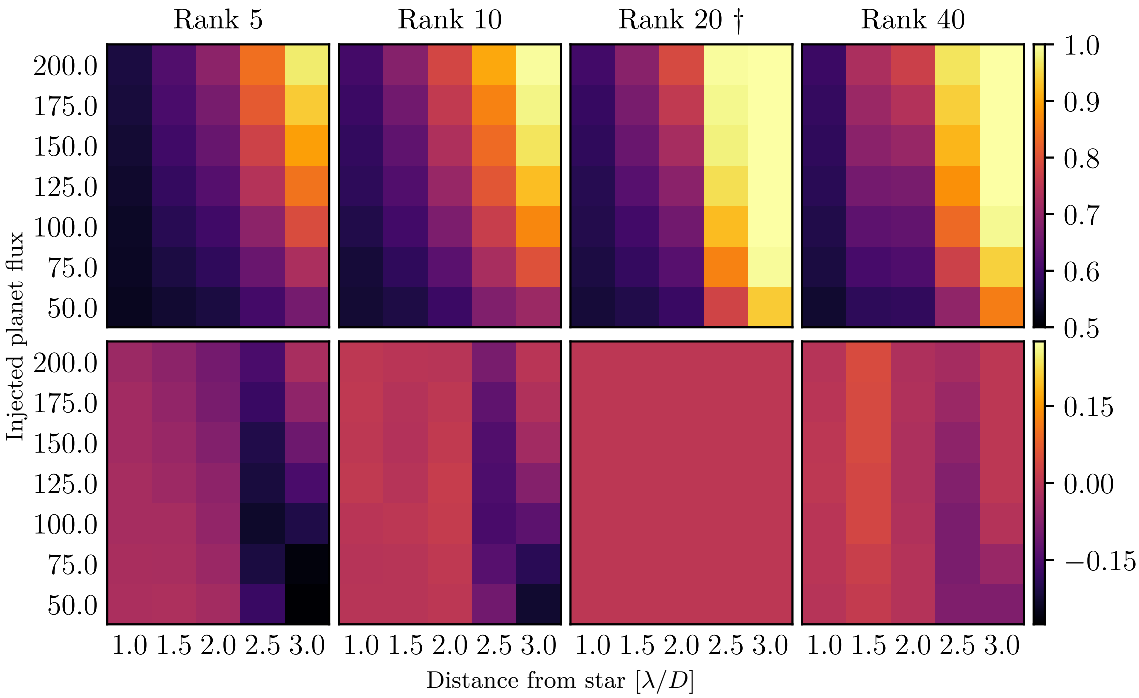

Figure 5.5: AUC (top) and differential (bottom) heatmaps of rank-increasing ISVHT for various ranks. The chosen reference heatmap is marked with a †.

40–80), performances near the star seem to drop. We attribute this to the fact that, close to the center, imputation cannot rely too much on temporal dependencies, because the trajectories move more slowly. Therefore, setting a high value of λ_w enforces a behavior that is ultimately damaging.

Results for the spatial regularizer also highlight an average improvement compared to a low-rank model without a spatial graph. However, increasing the value of the regularization factor seems to produce *better* models near the star. Once again, we attribute this to the fact that data is scarce near the center, due to the slow-moving trajectories. The difference is that, here, the information transfer enforced by the spatial graph fills in for the missing temporal data.

Finally, we observe that changing the temporal correlation length and spatial correlation radius has a similar impact as changing the regularization factors.

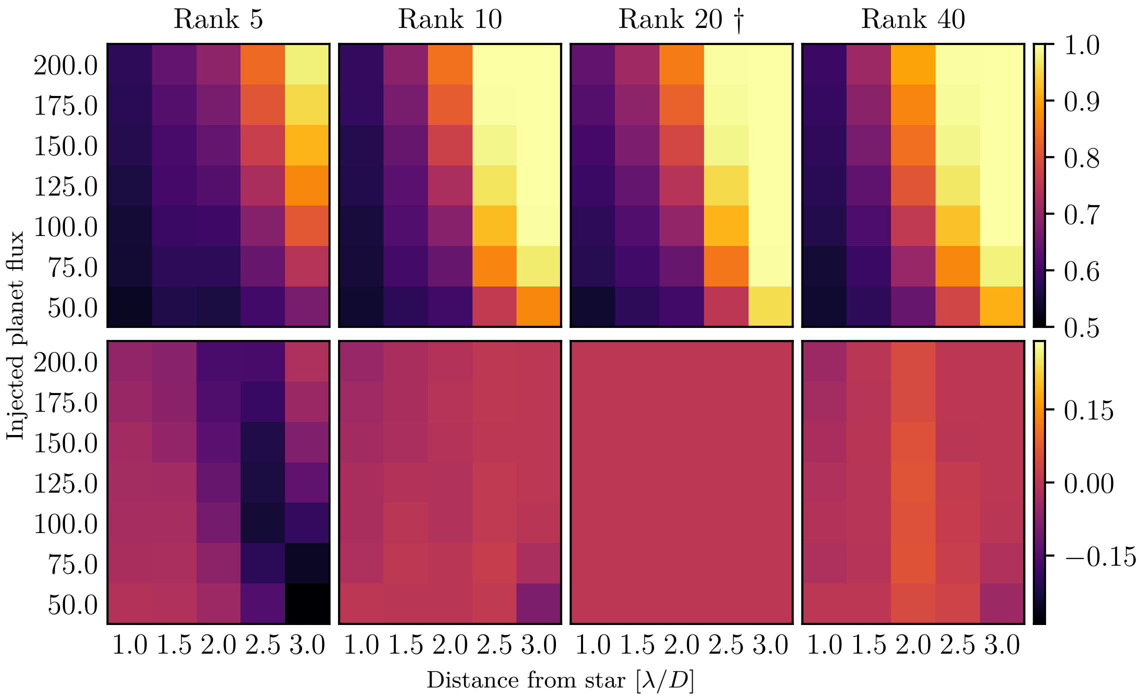

Figure 5.6: AUC (top) and differential (bottom) heatmaps of GRMF for various ranks. Other parameters are set to the values listed in Table 5.1. The chosen reference heatmap is marked with a †.

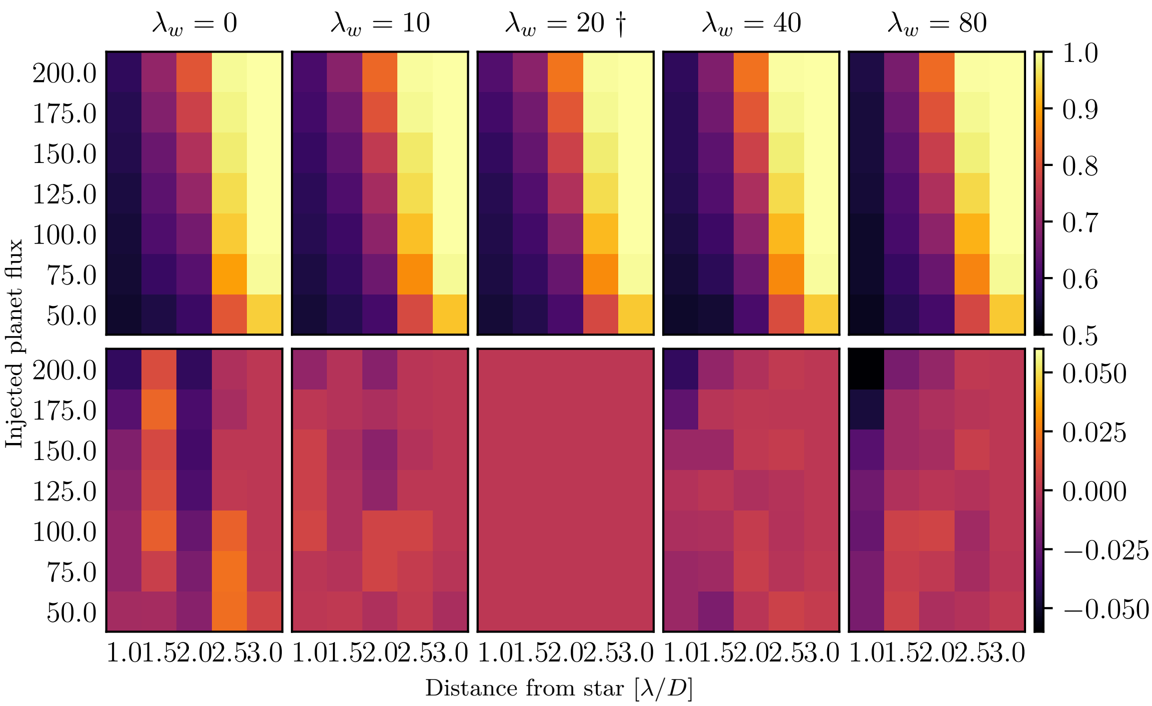

Figure 5.7: AUC (top) and differential (bottom) heatmaps of GRMF for various temporal regularization factors λ_w. The spatial graph is turned off ($\lambda_h = 0$) and other parameters are set to the values listed in Table 5.1. The chosen reference heatmap is marked with a †. In the leftmost column, the temporal graph is effectively turned off.

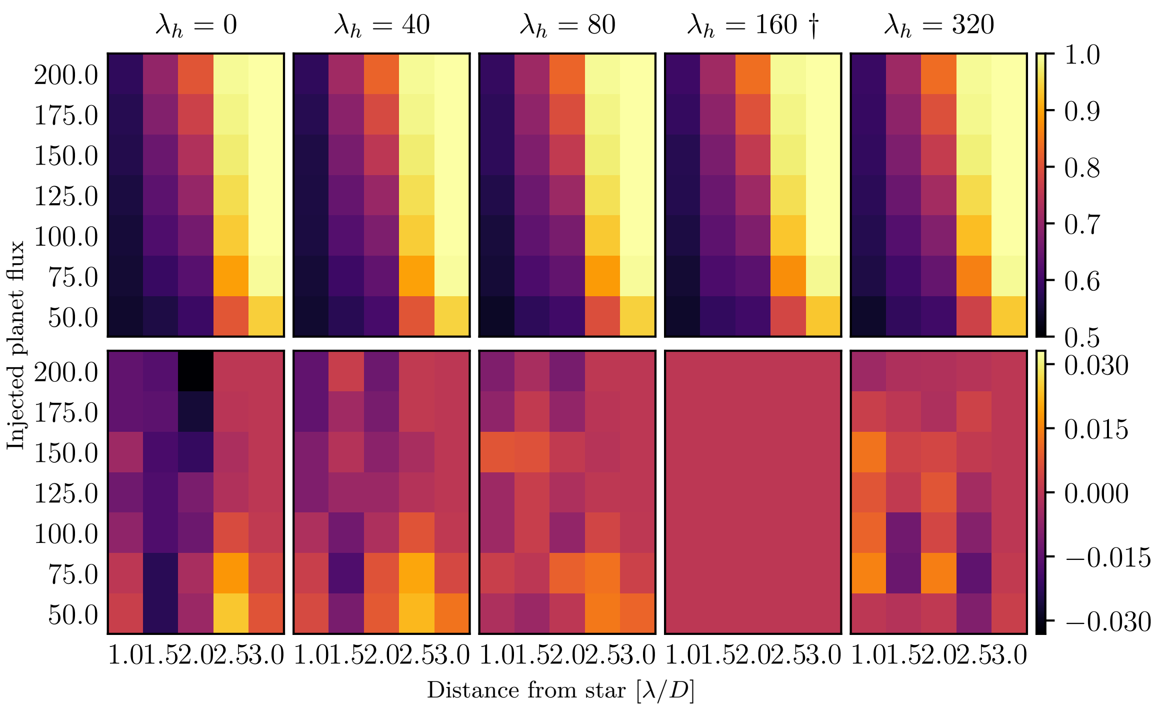

Figure 5.8: AUC (top) and differential (bottom) heatmaps of GRMF for various spatial regularization factors λ_h. The temporal graph is turned off ($\lambda_w = 0$) and other parameters are set to the values listed in Table 5.1. The chosen reference heatmap is marked with a †. In the leftmost column, the spatial graph is effectively turned off.

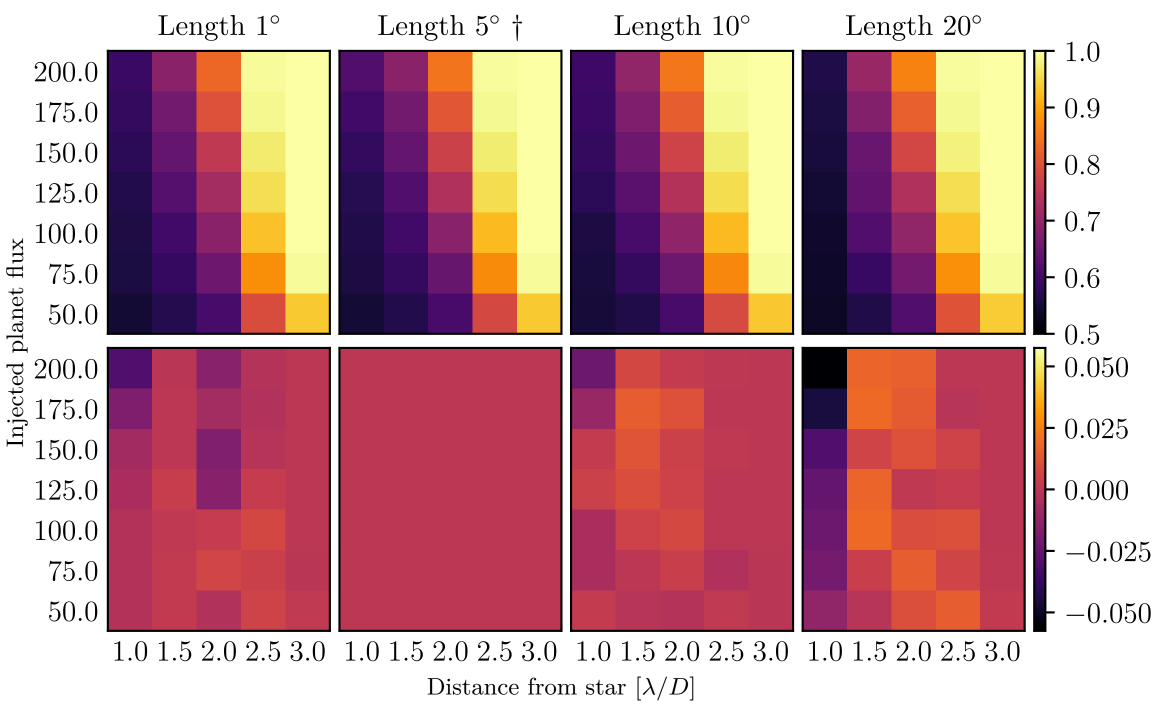

Figure 5.9: AUC (top) and differential (bottom) heatmaps of GRMF for various temporal correlation lengths. The spatial graph is turned off ($\lambda_h = 0$) and other parameters are set to the values listed in Table 5.1. The chosen reference heatmap is marked with a †.

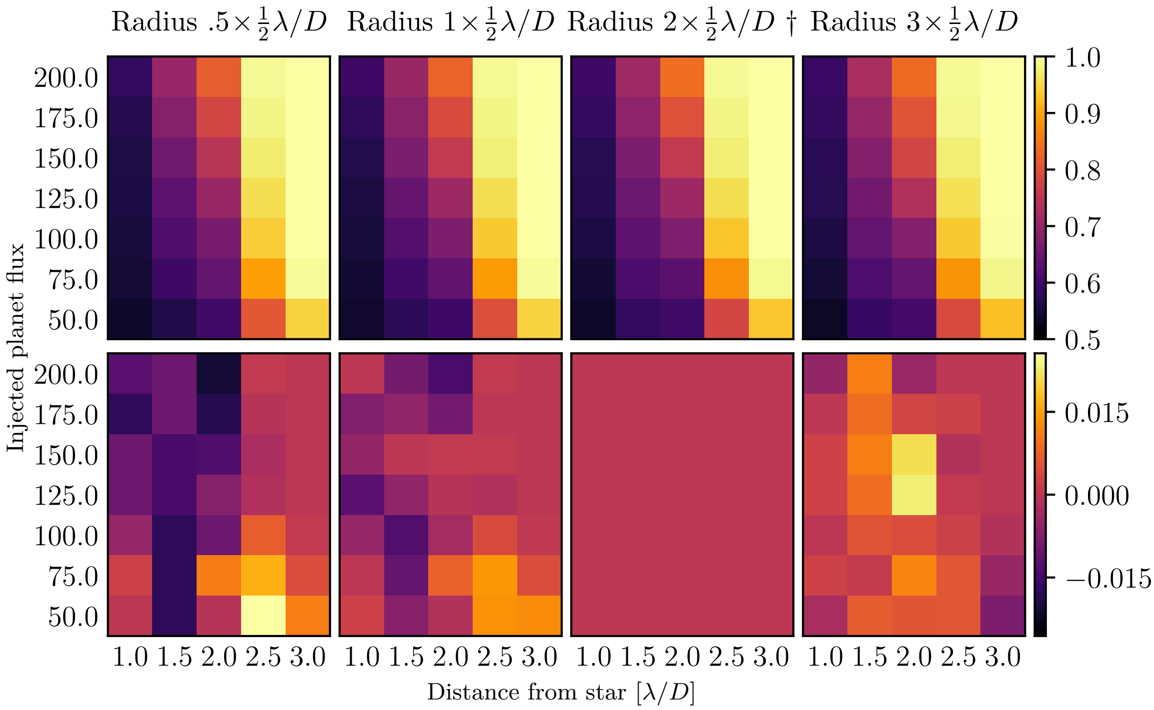

Figure 5.10: AUC (top) and differential (bottom) heatmaps of GRMF for various spatial correlation radii. The temporal graph is turned off ($\lambda_w = 0$) and other parameters are set to the values listed in Table 5.1. The chosen reference heatmap is marked with a †.

Chapter 6

Conclusion

6.1 Summary of this Thesis

We have introduced a new technique for PSF estimation and detection in angular differential imaging. This technique relies on matrix completion to construct a model of the PSF. We applied it with various matrix completion methods, and evaluated it against the state of the art PCA, in the framework of signal detection theory.

While the technique produced good results, it was ultimately inferior to PCA. However, we do not exclude that future improvements might change this. We also highlighted the potential of graph-regularized matrix factorization without completion, as a kind of PCA with graph prior.

6.2 Possible Improvements and Future Work

Matrix completion for PSF estimation. Here, we list a few possible leads for improving the matrix completion-based technique we presented.

- Parameters for the matrix completion methods were mostly hand-picked. Ideally, we would want some kind of cross-validation to select appropriate values. As a validation set, we can take the exact same cube, but with opposite parallactic angles, such that planetary signals no longer lie along the trajectories. Optimal parameters for the completion methods can then be selected according to how well they manage to reconstruct the original cube over the removed entries.

- The size of the aperture removed along the trajectory has a critical impact on the performance of this technique. In particular, when this aperture becomes too large, the quality of the model PSF plummets. The results presented in this document were acquired with a diameter of λ/D. Another size—possibly different at various radii—could maybe give better results.

- The technique presented here is applied to the full ADI cube, which takes a long time. It might be possible to achieve a significant speed-up by working on a subset of entries, *e.g.* an annulus. Another idea could be to run completion on more than one trajectory at a time. However, these changes are likely to affect the quality of detections. A detailed analysis would reveal whether this is the case.

- Another way to reduce the computational load is to subsample the cube in the spatial and/or temporal dimension. Once again, this is likely to have an impact on the detection performances, but this impact could be positive: When the cube is subsampled, the value (in pixels) of λ/D decreases too. Consequently, less pixels need to be imputed, and the pixels become more decorrelated.

GRMF. Graph-based models for PSF estimation look very promising. In this work, we only explored the surface of what is possible with this approach. Mostly, alternate ways of constructing the graphs deserve to be investigated.

- Currently, the spatial graph is based on the reference PSF. It might be interesting to proceed in a similar manner to the temporal graph, and assign the weights based on the empirical spatial correlation. This correlation could be computed annulus-per-annulus.
- Because the speckle noise behaves differently closer to the star, it might be interesting to estimate the temporal correlation separately for each λ/D annulus. This would require a small change to the current formulation of the GRMF problem (1.28).
- When looking at an ADI "video", speckles at very different locations appear to be somewhat correlated. This suggests we might get good results by using a non-local spatial graph, with greater weights for pixels that behave similarly.

Bibliography

Absil, O. et al. (2013). "Searching for companions down to 2 AU from β Pictoris using the L$'$-band AGPM coronagraph on VLT/NACO". In: *Astronomy & Astrophysics* 559, p. L12. ISSN: 0004-6361, 1432-0746. DOI: 10.1051/0004-6361/201322748.

Amara, Adam and Sascha P. Quanz (2012). "PYNPOINT: An image processing package for finding exoplanets". In: *Monthly Notices of the Royal Astronomical Society* 427.2, pp. 948–955. ISSN: 0035-8711. DOI: 10.1111/j.1365-2966.2012.21918.x.

Barrett, R. et al. (1994). *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition*. Philadelphia, PA: SIAM.

Beckers, J. M. (1993). "Adaptive optics for astronomy - Principles, performance, and applications". In: *\araa* 31, pp. 13–62. DOI: 10.1146/annurev.aa.31.090193.000305.

Blumensath, Thomas and Mike E. Davies (2009). "Iterative hard thresholding for compressed sensing". In: *Applied and Computational Harmonic Analysis* 27.3, pp. 265–274. ISSN: 1063-5203. DOI: 10.1016/j.acha.2009.04.002.

Bonse, Markus J., Sascha P. Quanz, and Adam Amara (2018). "Wavelet based speckle suppression for exoplanet imaging-Application of a de-noising technique in the time domain". In: *arXiv preprint arXiv:1804.05063*.

Borlinghaus, Rolf T. (2015). *Super-Resolution – On a Heuristic Point of View About the Resolution of a Light Microscope*. Leica Science Lab. URL: https://www.leica-microsystems.com/science-lab/super-resolution-on-a-heuristic-point-of-view-about-the-resolution-of-a-light-microscope/ (visited on 05/31/2018).

Buades, A., B. Coll, and J. Morel (2005). "A Review of Image Denoising Algorithms, with a New One". In: *Multiscale Modeling & Simulation* 4.2, pp. 490–530. ISSN: 1540-3459. DOI: 10.1137/040616024.

Cai, Jian-Feng, Emmanuel J. Candes, and Zuowei Shen (2008). "A Singular Value Thresholding Algorithm for Matrix Completion". In: *arXiv:0810.3286 [math]*. arXiv: 0810.3286.

Candes, Emmanuel J. and Yaniv Plan (2009). "Matrix Completion With Noise". In: *arXiv:0903.3131 [cs, math]*. arXiv: 0903.3131.

Candes, Emmanuel J. and Benjamin Recht (2008). "Exact Matrix Completion via Convex Optimization". In: *arXiv:0805.4471 [cs, math]*. arXiv: 0805.4471.

Candes, Emmanuel, Justin Romberg, and Terence Tao (2005). "Stable Signal Recovery from Incomplete and Inaccurate Measurements". In: *arXiv:math/0503066*. arXiv: math/0503066.

Cantalloube, F. et al. (2015). "Direct exoplanet detection and characterization using the ANDROMEDA method: Performance on VLT/NaCo data". In: *Astronomy & Astrophysics* 582, A89. ISSN: 0004-6361, 1432-0746. DOI: 10.1051/0004-6361/201425571. arXiv: 1508.06406.

Deeg, Hans J. and Roi Alonso (2018). “Transit Photometry as an Exoplanet Discovery Method”. In: *arXiv:1803.07867 [astro-ph]*. arXiv: 1803.07867.

Dohono, D. L. (1992). International Conference on Wavelets and Applications. Toulouse, France.

Dong, Shuyu, P.-A. Absil, and Kyle A. Gallivan (2018). “Graph learning for regularized low rank matrix completion”. In: *23rd International Symposium on Mathematical Theory of Networks and Systems (MTNS 2018)*.

Duchêne, Gaspard (2008). “High-angular resolution imaging of disks and planets”. In: *New Astronomy Reviews*. Circumstellar Disks and Planets at Very High Angular ResolutionProceedings of the VLTI Euro Summer School Held in Porto, Portugal, 28 May-8 June 2007 52.2, pp. 117–144. ISSN: 1387-6473. DOI: 10.1016/j.newar.2008.04.007.

Fazel, Maryam (2002). “Matrix rank minimization with applications”. PhD thesis. PhD thesis, Stanford University.

Fitzgerald, Michael P. and James R. Graham (2006). “Speckle Statistics in Adaptively Corrected Images”. In: *The Astrophysical Journal* 637.1, p. 541. ISSN: 0004-637X. DOI: 10.1086/498339.

Gomez Gonzalez, C. A., O. Absil, and M. Van Droogenbroeck (2017). “Supervised detection of exoplanets in high-contrast imaging sequences”. In: *Astronomy & Astrophysics*. ISSN: 0004-6361, 1432-0746. DOI: 10.1051/0004-6361/201731961.

Gonzalez, C. A. Gomez, O. Absil, et al. (2016). “Low-rank plus sparse decomposition for exoplanet detection in direct-imaging ADI sequences - The LLSG algorithm”. In: *Astronomy & Astrophysics* 589, A54. ISSN: 0004-6361, 1432-0746. DOI: 10.1051/0004-6361/201527387.

Gonzalez, C. A. Gomez, O. Wertz, et al. (2017). “VIP: Vortex Image Processing package for high-contrast direct imaging”. In: *The Astronomical Journal* 154.1, p. 7. ISSN: 1538-3881. DOI: 10.3847/1538-3881/aa73d7. arXiv: 1705.06184.

Jolliffe, I. T. (2002). *Principal Component Analysis*. 2nd ed. Springer Series in Statistics. New York: Springer-Verlag. ISBN: 978-0-387-95442-4.

Kuhn, J. R., D. Potter, and B. Parise (2001). “Imaging Polarimetric Observations of a New Circumstellar Disk System”. In: *The Astrophysical Journal Letters* 553.2, p. L189. ISSN: 1538-4357. DOI: 10.1086/320686.

Lovis, C. and D. Fischer (2010). “Radial Velocity Techniques for Exoplanets”. In: *Exoplanets*, pp. 27–53.

Lu, Yuan and Jie Yang (2015). “Notes on Low-rank Matrix Factorization”. In: *arXiv:1507.00333 [cs]*. arXiv: 1507.00333.

Lyot, Bernard (1939). “The Study of the Solar Corona and Prominences without Eclipses (George Darwin Lecture, delivered by M. Bernard Lyot, Assoc.R.A.S., on 1939 May 12)”. In: *Monthly Notices of the Royal Astronomical Society* 99.8, pp. 580–594. ISSN: 0035-8711. DOI: 10.1093/mnras/99.8.580.

Marois, Christian, David Lafrenière, René Doyon, et al. (2006). “Angular Differential Imaging: A Powerful High-Contrast Imaging Technique”. In: *The Astrophysical Journal* 641.1, p. 556. ISSN: 0004-637X. DOI: 10.1086/500401.

Marois, Christian, David Lafrenière, Bruce Macintosh, et al. (2008). “Confidence Level and Sensitivity Limits in High-Contrast Imaging”. In: *The Astrophysical Journal* 673.1, p. 647. ISSN: 0004-637X. DOI: 10.1086/523839.

Mawet, D. et al. (2014). "Fundamental Limitations of High Contrast Imaging Set by Small Sample Statistics". In: *The Astrophysical Journal* 792.2, p. 97. ISSN: 1538-4357. DOI: 10.1088/0004-637X/792/2/97.

Meka, Raghu, Prateek Jain, and Inderjit S. Dhillon (2009). "Guaranteed Rank Minimization via Singular Value Projection". In: *arXiv:0909.5457 [cs, math]*. arXiv: 0909.5457.

Oswalt, Terry and Ian S. McLean, eds. (2013). *Planets, Stars and Stellar Systems: Volume 1: Telescopes and Instrumentation*. Springer Netherlands. ISBN: 978-94-007-5620-5.

Pairet, Benoît et al. (2018). "STIM map: detection map for exoplanets imaging beyond asymptotic Gaussian residual speckle noise". In: *Monthly Notices of the Royal Astronomical Society*.

Racine, René et al. (1999). "Speckle Noise and the Detection of Faint Companions". In: *Publications of the Astronomical Society of the Pacific* 111, pp. 587–594. ISSN: 0004-6280. DOI: 10.1086/316367.

Rao, Nikhil et al. (2015). "Collaborative filtering with graph information: Consistency and scalable methods". In: *Advances in neural information processing systems*, pp. 2107–2115.

Saha, S. K. (2003). "Speckle Interferometric Technique for Astronomical Observations". In: *Journal of Optics* 32.1, pp. 37–56. ISSN: 0972-8821, 0974-6900. DOI: 10.1007/BF03354713.

Schneider, J. (2018). *Interactive Extra-solar Planet Catalog*. The Extrasolar Planets Encyclopedia. URL: http://exoplanet.eu/catalog/ (visited on 06/08/2018).

Schroeder, D. J. (2000). *Astronomical optics*. 2nd ed. San Diego: Academic Press. 478 pp. ISBN: 978-0-12-629810-9.

Soummer, Rémi, Laurent Pueyo, and James Larkin (2012). "Detection and Characterization of Exoplanets and Disks Using Projections on Karhunen-Loève Eigenimages". In: *The Astrophysical Journal Letters* 755.2, p. L28. ISSN: 2041-8205. DOI: 10.1088/2041-8205/755/2/L28.

Sprinthall, Richard C. (2011). *Basic Statistical Analysis (9th Edition)*. 9th ed. Pearson. ISBN: 978-0-205-05217-2.

Traub, Wesley A. and Ben R. Oppenheimer (2010). "Direct imaging of exoplanets". In: *Exoplanets*, pp. 111–156.

Vandenberghe, Lieven and Stephen Boyd (1996). "Semidefinite programming". In: *SIAM review* 38.1, pp. 49–95.

Wen, Zaiwen, Wotao Yin, and Yin Zhang (2012). "Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm". In: *Mathematical Programming Computation* 4.4, pp. 333–361.

Wolszczan, A. and D. A. Frail (1992). "A planetary system around the millisecond pulsar PSR1257 + 12". In: *Nature* 355, pp. 145–147. ISSN: 0028-0836. DOI: 10.1038/355145a0.

Zhou, Tianyi and Dacheng Tao (2011). "Godec: Randomized low-rank & sparse matrix decomposition in noisy case". In: *in International Conference on Machine Learning*.

— (2013). "Unmixing Incoherent Structures of Big Data by Randomized or Greedy Decomposition". In: *arXiv:1309.0302 [cs, stat]*. arXiv: 1309.0302.

Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve **www.uclouvain.be/epl**