A&A, 680, A86 (2023)
https://doi.org/10.1051/0004-6361/202346085

Astronomy & Astrophysics

NA-SODINN: A deep learning algorithm for exoplanet image detection based on residual noise regimes

C. Cantero[1,2], O. Absil[2,⋆], C.-H. Dahlqvist[2], and M. Van Droogenbroeck[1]

[1] Montefiore Institute, Université de Liège, 4000 Liège, Belgium
[2] STAR Institute, Université de Liège, Allée du Six Août 19C, 4000 Liège, Belgium
e-mail: ccantero@uliege.be

Received 6 February 2023 / Accepted 17 October 2023

ABSTRACT

Context. Supervised deep learning was recently introduced in high-contrast imaging (HCI) through the SODINN algorithm, a convolutional neural network designed for exoplanet detection in angular differential imaging (ADI) datasets. The benchmarking of HCI algorithms within the Exoplanet Imaging Data Challenge (EIDC) showed that (i) SODINN can produce a high number of false positives in the final detection maps, and (ii) algorithms processing images in a more local manner perform better.
Aims. This work aims to improve the SODINN detection performance by introducing new local processing approaches and adapting its learning process accordingly.
Methods. We propose NA-SODINN, a new deep learning binary classifier based on a convolutional neural network (CNN) that better captures image noise correlations in ADI-processed frames by identifying noise regimes. The identification of these noise regimes is based on a novel technique, named PCA-pmaps, which allowed us to estimate the distance from the star in the image from which background noise started to dominate over residual speckle noise. NA-SODINN was also fed with local discriminators, such as signal-to-noise ratio (S/N) curves, which complement spatio-temporal feature maps during the model's training.
Results. Our new approach was tested against its predecessor, as well as two SODINN-based hybrid models and a more standard annular-PCA approach, through local receiving operating characteristics (ROC) analysis of ADI sequences from the VLT/SPHERE and Keck/NIRC-2 instruments. Results show that NA-SODINN enhances SODINN in both sensitivity and specificity, especially in the speckle-dominated noise regime. NA-SODINN is also benchmarked against the complete set of submitted detection algorithms in EIDC, in which we show that its final detection score matches or outperforms the most powerful detection algorithms.
Conclusions. Throughout the supervised machine learning case, this study illustrates and reinforces the importance of adapting the task of detection to the local content of processed images.

Key words. techniques: image processing – methods: data analysis – methods: statistical – planets and satellites: detection – techniques: high angular resolution

1. Introduction

The direct imaging of exoplanets through 10-m class ground-based telescopes is now a reality of modern astrophysics (e.g. Bohn et al. 2021; Chauvin et al. 2017; Keppler et al. 2018; Marois et al. 2008b, 2010; Rameau et al. 2013; Wagner et al. 2016). Reaching this milestone is the result of significant advances in the field of high-contrast imaging (HCI). For instance, extreme adaptive optics (AO) is routinely used during observations to correct image degradation caused by the Earth's atmosphere (Snik et al. 2018). In the same way, dedicated HCI instruments, such as Subaru/SCExAO (Lozi et al. 2018) or VLT/SPHERE (Beuzit et al. 2019), make use of state-of-the-art coronagraphs (Soummer 2005; Mawet et al. 2009) in order to block out the starlight and mitigate the huge flux ratio (or contrast) between a host star and its companions. Despite all of these approaches, a high-contrast image is still affected by different additive sources of noise, such as photon noise associated with residual stellar light and thermal background emission, speckle noise associated with residual atmospheric turbulence, or residual aberrations that arise in the optical train of the telescope and instrument (Males et al. 2021). Speckles are scattered starlight blobs in the image that can mimic the expected signal of an exoplanet in both shape and contrast. Therefore, beyond dedicated instrumental developments, powerful image post-processing algorithms are needed to disentangle true companions from speckles. In order to help algorithms achieve this goal, different observing strategies have been proposed, the most popular being angular differential imaging (ADI, Marois et al. 2006). An ADI dataset consists of a sequence of high-contrast images acquired in pupil-stabilized mode, where the instrument de-rotator tracks the telescope pupil instead of the field, in such a way that the instrument and optics in the telescope stay aligned while the image rotates in time due to the Earth's rotation. As a result, speckles associated with the telescope and instrument optical train remain mostly fixed in the focal plane while the astrophysical signals rotate around the star as a function of the parallactic angle.

Currently, there exist a plethora of post-processing detection algorithms that work on ADI sequences. Most of these algorithms belong to the point spread function (PSF) subtraction family, which aims to model the speckle field and subtract it from each frame in the ADI sequence, derotate the residual images according to the parallactic angles, and finally collapse them into a final frame (Marois et al. 2008a), commonly referred to as a processed frame. Examples of these techniques are the locally optimized combination of images (LOCI, Lafreniere et al. 2007)

⋆ F.R.S.-FNRS Senior Research Associate.

and its variants TLOCI (Marois et al. 2014) and MLOCI (Wahhaj et al. 2015), principal component analysis (PCA, Soummer et al. 2012; Amara & Quanz 2012), the low-rank plus sparse decomposition (LLSG, Gomez Gonzalez et al. 2016), and the non-negative matrix factorization (NMF, Ren et al. 2018). PSF subtraction is usually followed by a detection algorithm, which can be either based on an signal-to-noise ratio (S/N) map (Mawet et al. 2014) or on a more recent technique, such as the standardized trajectory intensity mean (STIM, Pairet et al. 2019) or the regime-switching model (RSM, Dahlqvist et al. 2020). Another family of algorithms, based on an inverse problem approach, relies on directly modelling the expected planetary signal and tracking it along the ADI sequence. This is typically done by estimating the contrast of the potential planetary signal via a maximum likelihood estimation. Examples of these methods include ANDROMEDA (Cantalloube et al. 2015), the forward model matched filter (FMMF, Ruffio et al. 2017), the exoplanet detection based on patch covariances (PACO, Flasseur et al. 2018), or the temporal reference analysis of planets (TRAP, Samland et al. 2021). Recently, post-processing approaches based on supervised machine learning have emerged in HCI. Gomez Gonzalez et al. (2018) introduced the SODIRF and SODINN machine learning models, which are two binary classifiers that use a random forest and a convolutional neural network (CNN), respectively, to distinguish between companion signatures and residual noise in processed frames. Yip et al. (2020) trained a generative adversarial network with real data from the NICMOS camera (Hubble Space Telescope) to obtain a suitable dataset for training a CNN discriminative model to image companions. More recently, Gebhard et al. (2022) proposed a modified version of the half-sibling regression (Schölkopf et al. 2016) using a ridge regression with generalized cross-validation. Also, Flasseur et al. (2023) presented deep PACO, an adaptation of the PACO algorithm to supervised learning through a CNN architecture, which resulted in an improvement on both the detection and characterization of exoplanets.

A large fraction of these techniques was benchmarked in the context of the Exoplanet Imaging Data Challenge (EIDC, Cantalloube et al. 2020), the first platform designed for a fair and common comparison of post-processing algorithms for exoplanet detection and characterization in HCI. From the whole set of conclusions provided by the first EIDC phase (Cantalloube et al. 2020), we relied on two of them to motivate this paper. First, we observed that detection algorithms that exploit the local behaviour of image noise obtained the highest detection score in the challenge leaderboard. Second, we found that supervised machine learning algorithms produced a relatively high number of false positives, compared with more standard algorithms. Thereby, with the aim of enhancing the supervised machine learning models, for this study, we explored a new stratified noise approach, through which they can better exploit noise statistics in the ADI dataset. This approach relies on the existence of two noise regimes in the processed frame: a speckle-dominated residual noise regime close to the star, and a background-dominated noise regime further away. Our goal is to spatially identify these regimes in the processed frame through the study of their statistical properties, and then adapt the SODINN neural network to work separately in each of them in order to improve its detection performance. Therefore, in Sect. 2 we first revisit noise statistics in HCI and present a novel statistical method that allowed us to empirically delimit noise regimes in processed frames. Then, in Sect. 3, we introduce the noise-adaptive SODINN (or NA-SODINN) detection algorithm, a neural network architecture optimized to work on noise regimes. Our deep learning method was also fed with local discriminators, such as S/N curves, that contain additional physical-motivated features and help the trained model to better disentangle an exoplanet signature from speckle noise. In Sect. 4, NA-SODINN is evaluated through local ROC analysis using a series of ADI datasets obtained with various instruments. During the evaluation, NA-SODINN is benchmarked against other state-of-the-art HCI detection algorithms. Section 5 concludes the paper.

2. Noise regimes in processed ADI images

The term local is often used in image processing to describe a process applicable to a smaller portion of the image, such as the neighbourhood of a pixel, in which pixel values exhibit a certain amount of correlation. In HCI, defining image locality implies a good understanding of the physical information captured in the image. A common way to define locality is linked to the noise distribution along the image field of view. For example, after some pre-processing steps (including background subtraction), a high-contrast image is composed of three independent components: (1) residual starlight under the form of speckles; (2) the signal of possible companions; and (3) the statistical noise associated with all light sources within the field of view, generally dominated by background noise in infrared observations. In these raw images, exoplanets are hidden because starlight speckles and/or background residuals dominate at all angular separations, and act as a noise source for the detection task. According to their origin, starlight speckles can be classified as instrumental speckles (Hinkley et al. 2007; Goebel et al. 2016), which are generally long-lived and therefore referred to as quasi-static speckles, and atmospheric speckles, which have a much shorter lifetime (Males et al. 2021). Speckle intensity is known to follow a modified Rician probability distribution (Soummer et al. 2007). Here, the locality of the noise is driven by the distance to the host star (Marois et al. 2008a), which already gives an indication on how local noise will be defined in a processed image. Consequently, a large fraction of post-processing algorithms currently work and process noise on concentric annuli around the star. For example, the annular-PCA algorithm (Absil et al. 2013; Gomez Gonzalez et al. 2016) performs PSF subtraction with PCA on concentric annuli. Nevertheless, more sophisticated local approaches have recently been proposed in the literature. For instance, both the TRAP algorithm (Samland et al. 2021) and the half-sibling regression algorithm (Gebhard et al. 2022) take into account the symmetrical behaviour of speckles around the star when defining pixel predictors for the model.

In this section, we aim to introduce an alternative local processing, well-suited for the SODINN framework, as explained later in the paper, based on the spatial stratification of the processed frame into (at least) two noise regimes. For illustrative purposes, we make use, in this section, of two ADI sequences chosen from the set of nine ADI sequences used in the EIDC (Cantalloube et al. 2020); see Table A.1 for more information about the EIDC datasets. Our two ADI sequences, referred to as *sph2* and *nrc3*, were respectively obtained with the VLT/SPHERE instrument (Beuzit et al. 2019) and the Keck/NIRC-2 instrument (Serabyn et al. 2017). They have the advantage of not containing any confirmed or injected companions, which makes them appropriate for algorithm development and tests that rely on the injection of exoplanet signatures in the image.

2.1. Spatial noise structure after ADI processing

Performing PSF subtraction on each high-contrast image in an ADI sequence generates a sequence of residual images where speckle noise is significantly reduced and partly whitened (Mawet et al. 2014). After derotating these residual images based on their parallactic angle and combining them into a final frame, the remaining speckles are further attenuated and whitened. This final frame is commonly referred to as processed frame. Because of the different post-processing steps and the whitening operator that removes correlation effects, the S/N map technique (Mawet et al. 2014), the industry standard for exoplanet detection in processed frames, makes use of the central limit theorem to state that residual noise in processed frames follows a Gaussian distribution, an assumption that even today has not been proven experimentally. From practice, it is known that this Gaussian assumption leads to high false positive detection rates (Marois et al. 2008a; Mawet et al. 2014) since residual speckle noise in processed frames is never perfectly Gaussian, and still dominates at small angular separations. Pairet et al. (2019) found experimentally that the tail decay of residual noise close to the star is better explained by a Laplacian distribution than a Gaussian distribution. Later, Dahlqvist et al. (2020) reached the same conclusion by applying a Gaussian and a Laplacian fit to the residuals of PCA-, NMF-, and LLSG-processed frames. These experimental results suggest the presence of two residual noise regimes in the processed frame: a non-Gaussian noise regime close to the star, dominated by residual speckle noise, and a Gaussian regime further away, dominated by background noise.

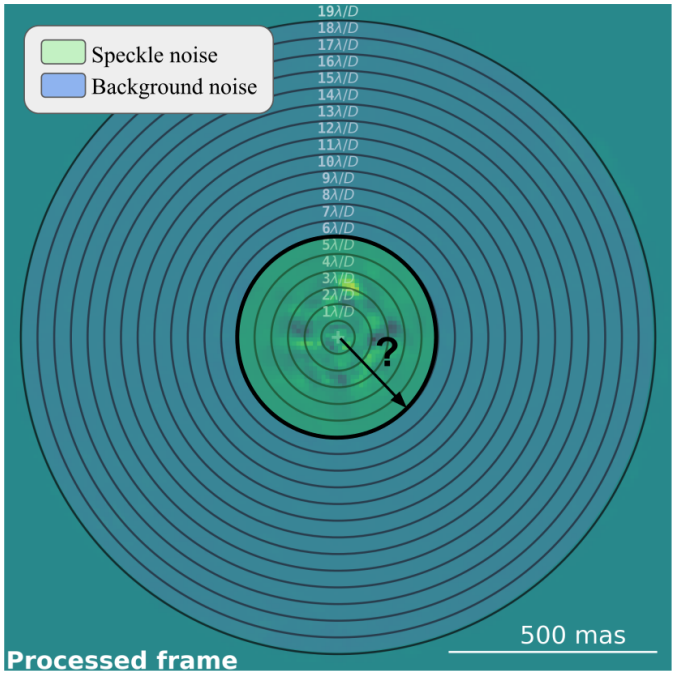


Fig. 1. Processed frame from *sph2* dataset with both speckle-dominated and background-dominated residual noise regimes and their annular split (black circle). The best approximation of this split is what we aim to find in this section.

2.2. Identification of noise regimes

Based on our understanding of the local statistics of noise in a processed frame, we aim now to spatially delimit both noise regimes in the image. To do so, we try to find the best radial distance approximation from the star where residual speckle noise starts to become negligible compared to background noise (Fig. 1), which is assumed to be uniform over the whole field of view.

2.2.1. Paving the image field of view

In order to find the radius at which background noise starts to dominate in the image, we study the evolution of noise statistics as a function of angular separation. We first pave the full field of view with concentric annuli of λ/D width (Fig. 1). Each annulus contains pixels that are expected to be drawn from the same parent population (Marois et al. 2008a), although we acknowledge that this working hypothesis cannot be completely fulfilled when diffraction patterns associated with the spiders of the telescope or the wind-driven halo are present in the image (Cantalloube et al. 2019). We note that, in the presence of residual speckles, pixels that contain information from the same speckle are all spatially correlated. When background noise dominates over residual speckle noise, we can instead assume that all pixels in an annulus are independent, since photon noise occurs on a pixel-wise basis. However, this assumption of independence can be non-optimal when bad pixels are interpolated, since it can still leave spatially correlated footprints. In HCI, a common procedure to guarantee the independence of samples when performing statistical analysis is to work by integrating pixel intensities on non-overlapping circular apertures of λ/D diameter within a given annulus (Mawet et al. 2014), as shown in Fig. 2. This

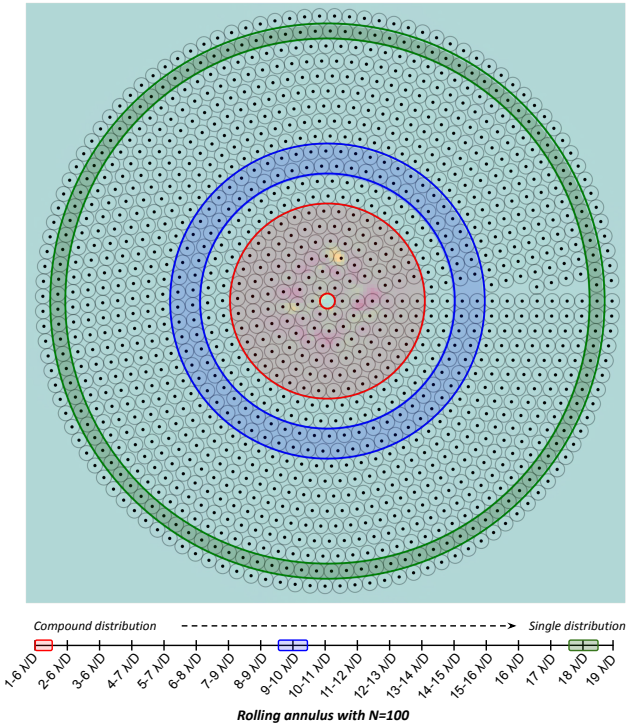


Fig. 2. Rolling annulus with $N = 100$ over the processed frame of Fig. 1. Top: examples of the first rolling one (in red), the ninth rolling one (in blue) and the eighteenth rolling one (in green) displayed over the central pixel pavement in the image. Bottom: a complete set of rolling annuli shown in a straight line that represents the distance from the star. The three rolling annuli shown in the top figure are displayed with the same colours.

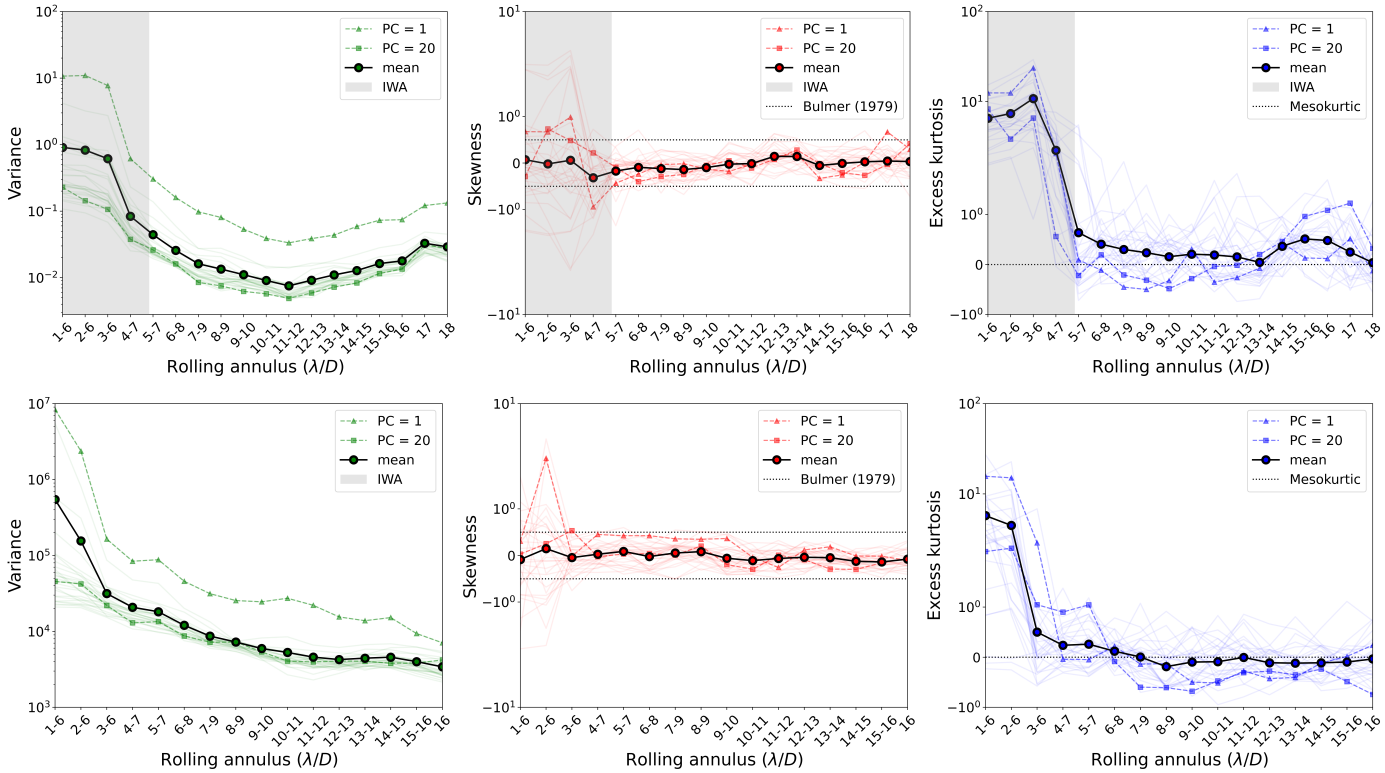


Fig. 3. Statistical moments evolution based on a rolling annulus which paves the full annular-PCA processed frame. The *top and bottom rows* refer, respectively, to the *sph2* and *nrc3* ADI sequences. Colour curves on each subplot refer to a different principal component ranging from one to thirty. The bold curve on top of each subplot indicates the average of the thirty PCs, and PC=1 and PC=20 are illustrated with specific symbols. In the case of *sph2*, grey areas highlight the inner working angle (IWA).

procedure is based on the characteristic spatial scale of residual speckles ($\sim\lambda/D$ size). However, Bonse et al. (2022) have recently showed that, in the presence of speckle noise, this independence assumption on non-overlapping apertures is incorrect. Instead, they propose to (i) only consider the central pixel value in each circular aperture to produce a more statistically independent set of pixels and (ii) possibly repeat the experiment with various spatial arrangements of the non-overlapping apertures to reduce statistical noise in the measured quantities. We follow this recommendation, and therefore, for the rest of this study, we define our annulus samples of both speckle- and background-dominated noise regimes by only taking the central pixel value for each non-overlapping circular aperture (Fig. 2). This approach also minimizes the possible effect of bad pixel interpolation.

One limitation in using non-overlapping apertures is the small sample statistics problem, especially at small angular distances (Mawet et al. 2014). Small samples generally lead to conclusions that are not strong enough statistically speaking. In order to avoid this issue, we propose to use the concept of a rolling annulus (Fig. 2) that always contains a minimum number of independent pixels N. These N pixels are the central pixels of apertures that pave the field of view and are included in the rolling annulus. It can be understood as an annular window around the star for which the inner boundary moves in $1\lambda/D$ steps, while the outer boundary is set to achieve the criterion on the minimum number of independent pixels. An example of this process with $N = 100$ pixels is shown in Fig. 2, where the first rolling annulus that achieves the condition, composed of all central pixels of the non-overlapping apertures between 1 and $6\lambda/D$, is displayed in red over the processed frame. Then, the rolling annulus moves away from the star changing its boundaries, as illustrated with the black line at the bottom of Fig. 2. For example, the ninth rolling annulus (in blue) with $N = 100$ is located between 9 and 10 λ/D, and the eighteenth rolling annulus (in green) is at $18\lambda/D$ distance, achieving the $N = 100$ condition without the need to expand the region to another annulus. In this paper, we select $N = 100$ minimum samples, considered to be the minimum number of samples required to reach reliable statistical power and significance for our statistical analysis.

2.2.2. Statistical moments

Once the processed frame is paved, we first study the evolution of different statistical moments, such as the variance (amount of energy), the skewness (distribution symmetry) and the excess kurtosis (distribution tails), as a function of the angular separation from the star for different number of principal components (PCs), ranging from component one to thirty. Figure 3 shows this evolution for the case of the *sph2* (top row) and *nrc3* (bottom row) datasets, on which we applied annular-PCA to produce the processed frames. We observe that the variance decreases as the rolling annulus moves away from the star. This trend is common to both datasets and is what we would expect in physical terms, as the intensity of residual speckles varies rapidly with angular separation, especially at short distances. We also see that this behaviour is dampened when using a larger number of principal components, which leads to more effective speckle subtraction. Regarding the skewness analysis, we adopt the convention of Bulmer (1979), which states that a distribution is symmetrical when its skewness ranges from −0.5 to 0.5. For both datasets, we clearly observe a loss of symmetry at small angular separations. The presence of speckles can provoke this distribution

asymmetry due to their higher intensity values in comparison with the background. Looking now at the excess kurtosis in Fig. 3, we observe a strong leptokurtic[1] trend for the entire set of PCs at small angular separations and for both datasets. This perfectly matches the fact that a Laplacian distribution fits better the tail decay of residual noise (Pairet et al. 2019), since it is, by definition, leptokurtic. At higher angular separations, instead, we observe differences between both datasets. In the *sph2* processed frames, we detect one mesokurtic regime approximately between 6 and 13λ/D followed by a weaker leptokurtic regime approximately between 14 and 18λ/D. For *nrc3*, we only observe one mesokurtic regime at a large distance from the star, beyond the third rolling annuli (Fig. 3).

2.2.3. Combined normality test analysis

Another way to explore the spatial distribution of noise is to use hypothesis testing. Assuming that residual speckle noise is non-Gaussian by nature, while background noise is Gaussian (see Sect. 2.1), we can assess the probability of the null hypothesis H_0 that data are normally distributed, that is, explained solely by background noise. We rely on a combination of a series of normality tests, making use of four of the most powerful tests: the Shapiro-Wilk test (*sw*, Shapiro & Wilk 1965), the Anderson-Darling test (*ad*, Anderson & Darling 1952), the D'Agostino-K2 test (*ak*, D'Agostino & Pearson 1973), and the Lilliefors test (*li*, Lilliefors 1967). This choice is motivated by the fact that they have been well-tested in many studies, including Monte-Carlo simulations (Yap & Sim 2011; Marmolejo-Ramos & González-Burgos 2013; Ahmad & Khan 2015; Patrício et al. 2017; Wijekularathna et al. 2019; Uhm & Yi 2021). It is worthwhile to remark that the goal is not to benchmark the robustness of all these tests. Our purpose, instead, is to collect a larger amount of statistical evidence for the same hypothesis, that can then be combined to increase the statistical power when making a decision regarding the null hypothesis. Moreover, regarding the statistical requirements, the only constraints to be verified before using these tests are the independence and sufficient size of the sample. In terms of sample size, Jensen-Clem et al. (2017) shows that normality tests can exhibit lower statistical power with sample sizes under 100 observations. Here, the independence and size constraints are met by the proposed approach to pave the field of view, using the central pixels of non-overlapping apertures within rolling annuli of $N = 100$ apertures. Additionally, we follow for this analysis the recommendation of Bonse et al. (2022) to perform our statistical tests with various spatial arrangements for the non-overlapping apertures. We leverage the fact that different aperture arrangements within the same annulus contain valuable noise diversity that can directly benefit the analysis when making a decision about the null hypothesis.

Our analysis for testing the null hypothesis H_0 within a specific rolling annulus of the processed frame is thus composed as follows. We begin by randomly selecting a normality test t from the set $\mathcal{T} = \{sw, ad, ak, li\}$. Subsequently, we randomly choose an angular displacement θ of circular apertures for each single annulus within the rolling annulus. Assuming N_{ann} single annuli, then, $\Theta = \{\theta_i\}_{i=\{1,\dots,N_{ann}\}}$, where Θ thus represents a random aperture arrangement. After defining the sample of central pixels $X(\Theta)$ for this arrangement, we use the selected statistical test t to compute the p-value associated to $X(\Theta)$. We denote this p-value as $p(t, \Theta)$. This process of randomly selecting both a normality test t and an aperture arrangement $X(\Theta)$ to compute $p(t, \Theta)$ is repeated m times for the same rolling annulus, which produces m p-values that are not statistically independent. The final step involves using the harmonic mean, as proposed by Vovk & Wang (2020), to combine these m p-values into a global p-value noted $\bar{p}$. By comparing $\bar{p}$ with a predefined significance threshold α, we can finally reject the null hypothesis H_0 if $\bar{p} < \alpha$.

By repeating this procedure for each rolling annulus in the processed frame and for various numbers of principal components in our annular-PCA post-processing algorithm, we can build what we call the PCA p-value map, or PCA-pmap for short. Figures 4a and 5a show examples of PCA-pmaps for the *sph2* and *nrc3* datasets, respectively. For both, we only considered the first 29 principal components to produce the annular-PCA space (y-axis in figures). Each cell in a PCA-pmap shows, through the number in white and its background colour, the combined p-value $\bar{p}$ with $m = 300$. P-values below the pre-defined threshold α are marked with yellow stars on the figures. In order to minimize the Type I error (false rejection of the null hypothesis), we selected the standard threshold value $\alpha = 0.05$ in Figs. 4a and 5a. Afterward, we calculate the fraction of yellow star markers, or H_0 rejection rate, along the PCA domain for each rolling annulus in PCA-pmaps. We then classify rolling annuli as speckle-dominated when they contain more than 50% of stars. Figures 4b and 5b show this selection criterion by plotting the H_0 rejection rate per rolling annulus. In the case of *sph2* (Fig. 4), we clearly observe the presence of four noise regimes beyond the inner working angle: a first regime dominated by non-Gaussian noise due to residual speckles between 5 and 7 λ/D distance, a second regime where noise is more consistent with Gaussian statistics, probably dominated by background noise between 8 and 14 λ/D, a third regime with non-Gaussian noise between 15 and 16 λ/D, where speckles are dominating again as we approach the limit of the well-corrected area produced by the SPHERE adaptive optics (Cantalloube et al. 2019), and finally, a fourth regime more consistent with Gaussian statistics again between 17 and 19 λ/D. The speckle-dominated regime at 15–16 λ/D would also explain the slightly leptokurtic behaviour observed at those separations in Fig. 3. For the *nrc3* dataset (Fig. 5), we only observe two noise regimes, with speckle noise dominating approximately between 1 and 3 λ/D distance, and background noise dominating beyond 3λ/D (Fig. 5b). The white dotted line and circles overplotted on the PCA-pmaps will be explained later in Sect. 3.2.

2.3. Field-of-view splitting strategy

At this point, we can see that, for both *sph2* and *nrc3*, similar estimations of the noise regimes are reached using the two proposed methods: the study of statistical moments and the PCA-pmaps. Figure 3 provides a first insight into the spatial structure of residual noise and, thereby, brings us closer to estimating the radius split (Fig. 1) in the processed frame. Indeed, the significant increase of the variance together with the leptokurtic behaviour and the positively skewed trend at small angular separations, suggest that this regime is still dominated by residual starlight speckles. On the other hand, PCA-pmaps contain more statistical diversity through the combination of p-values with which very similar regime estimations are reached. Thus, both analyses are complementary from a statistical perspective. Yet, from now on, we elect to use PCA-pmaps to define the noise regime as a baseline, since they can also be used for other purposes.

[1] In statistics, a leptokurtic distribution has a kurtosis greater than the kurtosis of a normal distribution (mesokurtic), and it is associated in HCI to increase the false alarm rate.

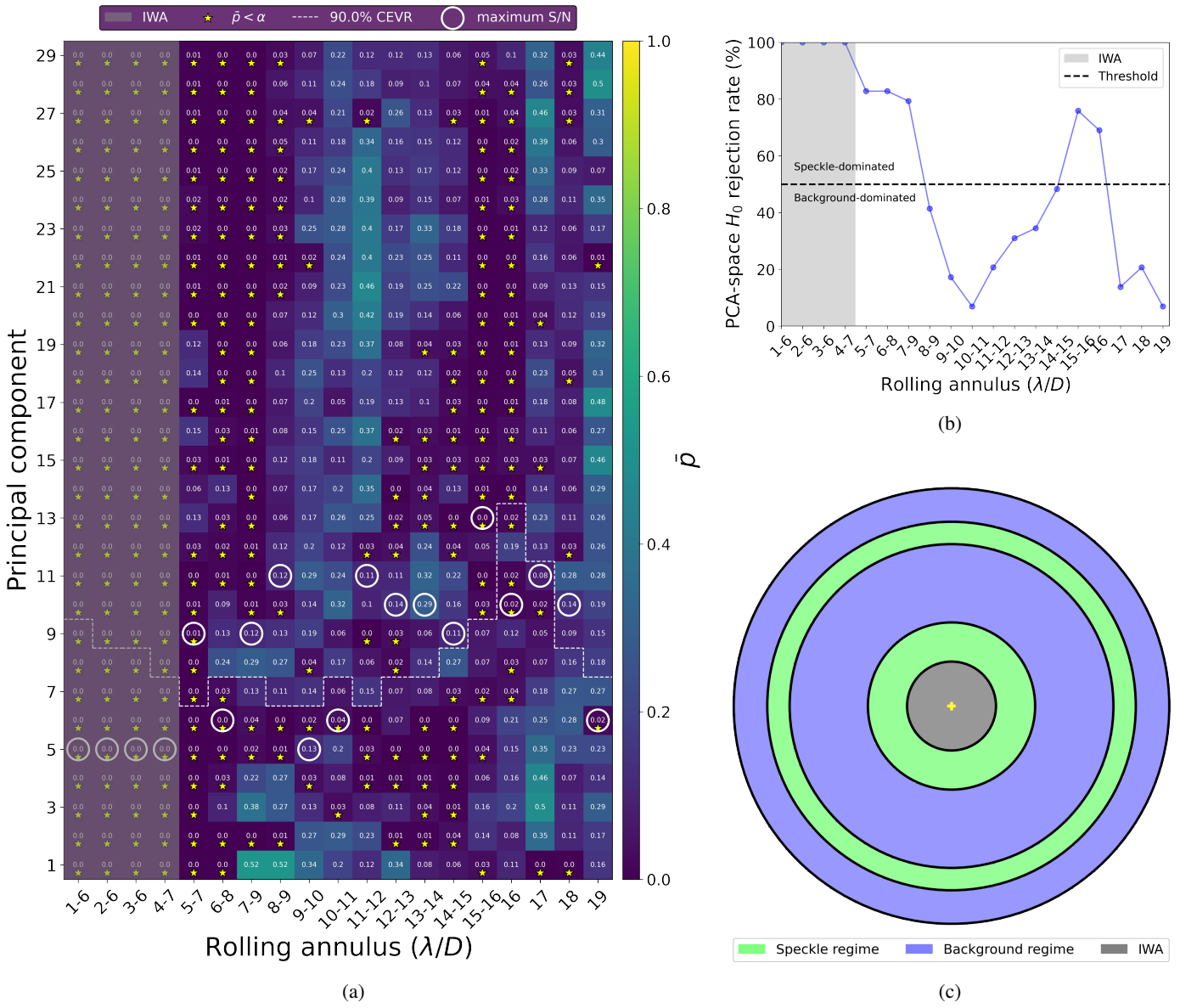


Fig. 4. Combined normality test analysis for the *sph2* ADI sequence. (*a*) PCA-pmap showing the combined p-value $\bar{p}$ both as a colour code and as values, as a function of the distance to the star through the rolling annulus (x axis) and the number of principal components used in the PCA-based PSF subtraction (y axis). Yellow star markers indicate when the null hypothesis H_0 (Gaussian noise) is rejected. The white dashed line shows the 90% CEVR at each rolling annulus. White circles in bold highlight the principal component that maximizes the S/N of fake companion recoveries. (*b*) Percentage of yellow star markers, or H_0 rejection, (y axis) for each rolling annulus (x axis) on the PCA-pmap. The dashed black line highlights the selection criteria for setting the dominant noise at each rolling annulus. (*c*) Final representation of estimated noise regimes along the processed frames field of view. Grey areas in each subplot highlight the inner working angle (denoted as IWA).

The noise analysis described above suggests that noise regions should be defined on a case-by-case basis. Regarding the nature of residual noise in a processed frame, our tests do not necessarily mean that residual speckle noise is non-Gaussian in the innermost, individual annuli. Instead, compound distributions could be at the origin of the non-Gaussian noise behaviour in the innermost rolling annuli. Compound distributions refer to the sampling of random variables that are not independent and identically distributed. For small angular separations (red annulus in Fig. 2) where residual speckle noise dominates over background noise, the samples are taken from distributions that might be Gaussian, but with different variances. If they are Gaussian and their variance follows an exponential distribution, then according to Gneiting (1997), the compound distribution follows a Laplacian, as observed by Pairet et al. (2019). This explanation, which is not a proof, would reconcile the belief that residual speckle noise should be locally Gaussian. Because of the small sample size, there is, however, no proper way to test this interpretation on individual annuli in the innermost regions. For all these reasons, we believe that splitting the processed frame field of view in different noise regimes is duly motivated, and, in the next sections, we detail how we have implemented this splitting to improve the detection of exoplanets.

3. Implementation

So far, we have focused on understanding the spatial structure of residual noise in the processed frame, which has allowed us to

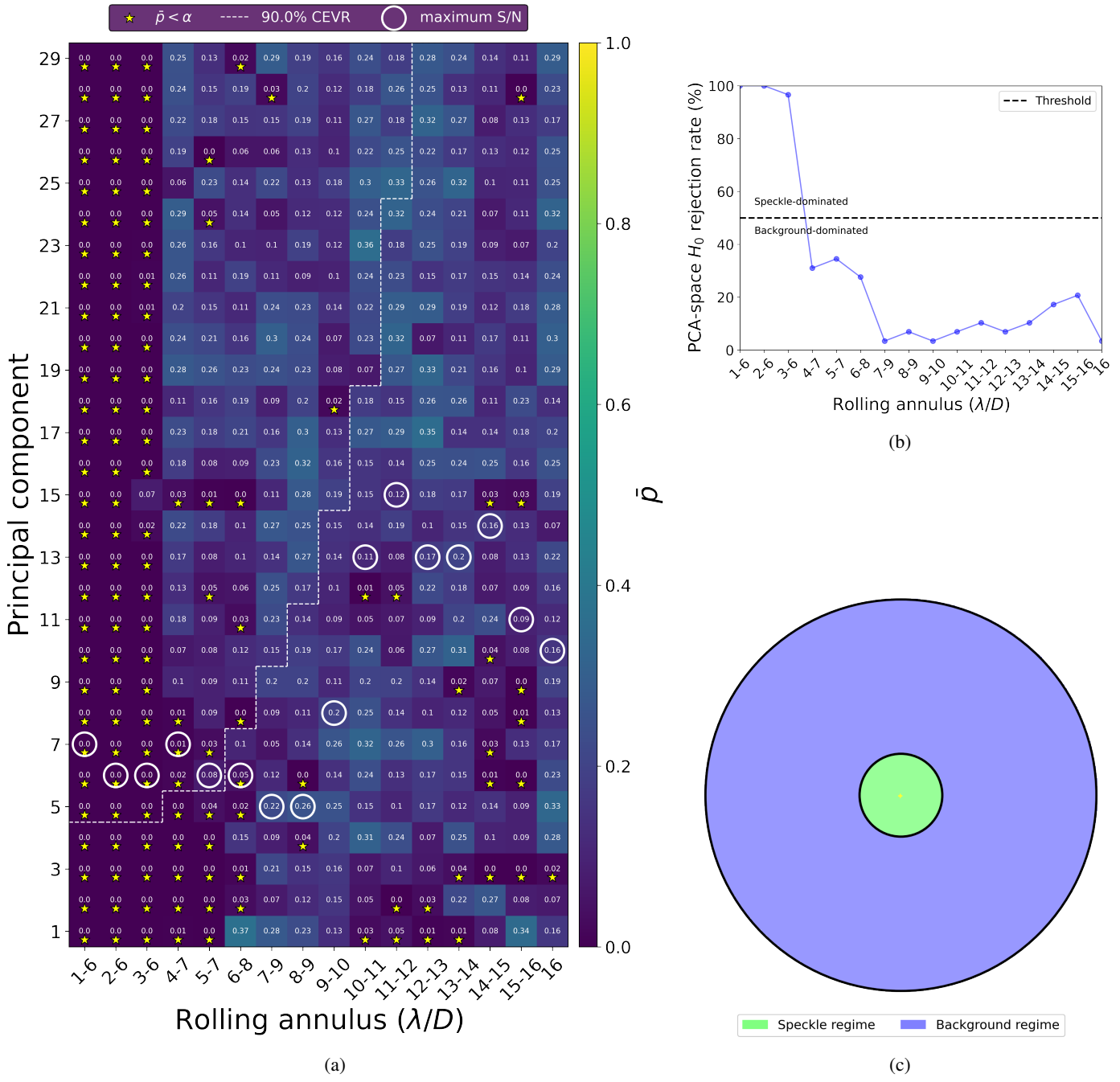


Fig. 5. Same as Fig. 4, but for the *nrc3* ADI sequence.

empirically define the regions dominated by speckle and background noise. Now, we aim to use this local noise approach in order to help post-processing algorithms enhance their detection performance. Most HCI algorithms have the potential of being applied separately to different noise regimes. Here, we are particularly interested in the case of deep learning. Neural networks are good candidates to capture image noise dependencies due to their ability to recognize hidden underlying relationships in the data and make complex decisions. In order to maximize the added value of working in noise regimes and showing its benefits for the detection task, we propose to revisit SODINN (Gomez Gonzalez et al. 2018), the first supervised deep learning algorithm for exoplanet imaging. In this section, we first provide a brief overview of SODINN, and then present our novel NA-SODINN algorithm, an adaptation of SODINN working on noise regimes, aided with additional handcrafted features.

3.1. Baseline model: The SODINN algorithm

SODINN stands for Supervised exOplanet detection via Direct Imaging with a deep Neural Network. It is a binary classifier that uses a convolutional neural network (CNN) to distinguish between two classes of square patch sequences: sequences that contain an exoplanet signature (c_+, the positive class), and sequences that contain only residual noise (c_-, the negative class). Figure 6 (bottom) shows an example sequence for each class, where the individual images are produced with various numbers of principal components. The first image in the

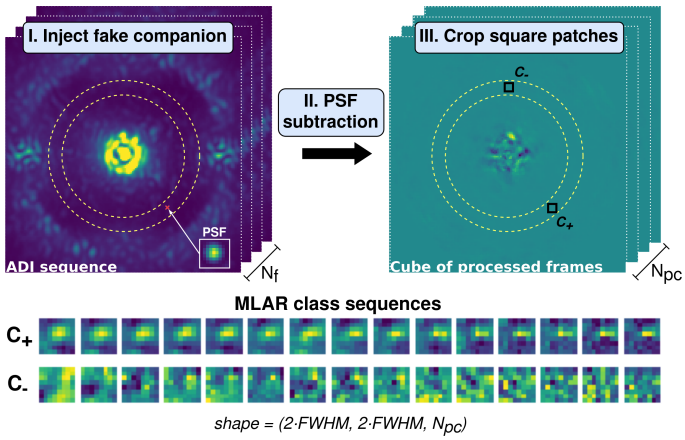


Fig. 6. SODINN labelling stage. *Top*: steps for generating MLAR samples (see the text for more details). N_f is the number of frames in the ADI , and N_{pc} is the number of principal components in the cube of processed frames and therefore in the final MLAR sequence. *Bottom*: example of an MLAR sequence of each class.

sequence corresponds to the first principal component, while the last corresponds to a number of principal components with which a maximum of 90% cumulative explained variance ratio (CEVR) is captured. Gomez Gonzalez et al. (2018) refer to these patch sequences as Multi-level Low-rank Approximation Residual (MLAR) samples.

3.1.1. Generation of the training set

The first step in SODINN is to build a training dataset composed of thousands of different c_+ and c_- MLAR sequences. A c_+ sequence is formed through three consecutive steps that are summarized in Fig. 6. (i) First, a PSF-like source is injected at a random pixel within a given annulus of the ADI sequence. The flux of this injection is the result of multiplying the normalized off-axis PSF by a scale factor randomly chosen from a pre-estimated flux range that corresponds to a pre-defined range of S/N in the processed frame. The estimation of injection flux ranges is explained in Appendix B. (ii) Singular value decomposition (SVD, Halko et al. 2011) is then used on this synthetic ADI sequence to perform PSF subtraction for different numbers of singular vectors (or principal components), thereby producing a series of processed frames. (iii) Finally, square patches are cropped around the injection coordinates for each processed frame. This forms a series of c_+ MLAR sequences, where each sequence contains the injected companion signature for different numbers of principal components. The patch size is usually defined between 1.5 and 2 times the FWHM of the PSF.

Likewise, we construct a c_- sequence by extracting MLAR sequences for pixels where no fake companion injection is performed. The number and order of singular vectors are the same as those used for the c_+ sequences. For the case of c_- sequences, SODINN must deal with the fact that, using only one ADI sequence, we obtain a single realization of the residual noise, so that the number of c_- sequences we can grab per annuli is not enough to train the neural network without producing overfitting. SODINN solves this problem by increasing the number of c_- sequences in a given annulus through a dedicated data augmentation strategy that is based on four consecutive steps: (i) build a first subset by randomly grabbing c_- sequences centred on up to ten percent of the total number of pixels; (ii) build a second subset by grabbing all the available pixels in the annulus and flip the sign of the parallactic angle when derotating the residual images, a common practice in HCI to remove possible planetary sources while preserving noise properties; (iii) randomly pick groups of three c_- sequences from the two subsets and average them to produce new sequences; (iv) finally, perform random rotations and small shifts of the c_- sequences obtained in the previous step to create even more diversity. The same rotation angle and shift are applied to all the slices of a given MLAR sequence. This data augmentation process ensures that we only use augmented c_- sequences for the training.

This procedure of generating c_+ and c_- sequences is repeated thousands of times for each annulus in the field of view. When the entire field of view is covered, MLAR sequences of the same class from all annuli are mixed, and the balanced training set (same amount of c_+ and c_- samples) is built.

3.1.2. Training of the network

The training set is then used to train the SODINN neural network. This produces a detection model that is specific for the ADI sequence from where MLAR sequences were generated. The SODINN network architecture is composed of two concatenated convolutional blocks. The first block contains a convolutional-LSTM (Shi et al. 2015) layer with 40 filters and a hyperbolic tangent activation function, and kernel and stride size of (1,1), followed by a spatial 3D dropout (Srivastava et al. 2014) and a MaxPooling-3D (Boureau et al. 2010). The second block contains the same except that it now has 80 filters, and kernel and stride size of (2, 2). These first two blocks extract the feature maps, capturing all spatio-temporal correlations between pixels of MLAR sequences. After that, they are flattened and sent to a fully connected dense layer of 128 hidden units. Then, a rectifier linear unit (ReLU, Nair & Hinton 2010) is applied to the output of this layer followed by a dropout regularization layer. Finally, the output layer of the network consists of a sigmoid unit, which provides a normalized value between 0 and 1. This value is usually referred as a probability, however, it is known in computer vision that the output of a deep learning architecture normalized between 0 and 1 with classical activation functions (e.g. the sigmoid function) tends to be more binary, and therefore, it cannot be interpreted as a real probability. For this reason, from now on, we refer to this output score as the model confidence. The network weights are initialized randomly using a Xavier uniform initializer, and are learned by back-propagation with a binary cross-entropy cost function:

$$L(y_n, \hat{y_n}) = -\sum_n (y_n \ln(\hat{y}_n) + (1 - y_n) \ln(1 - \hat{y}_n)), \quad (1)$$

where y_n is the true label of the n^{th} MLAR sample and $\hat{y}_n$ is the predicted confidence that this n^{th} MLAR sample belongs to the c_+ class. SODINN uses an Adam optimizer with a step size of 0.003, and mini-batches of 64 training samples. An early stopping condition monitors the validation loss. The number of epochs is usually set to 15, with which SODINN generally reaches ~99.9% validation accuracy (Gomez Gonzalez et al. 2018).

3.1.3. Inference

Once the detection model is trained and validated, it is finally used to find real exoplanets in the same ADI sequence. Because the input of the model is an MLAR structure, we first map the

entire field of view by creating MLAR samples (with no injection) centred on each pixel. These MLAR samples have never been processed during the training since the c_- class MLAR samples in the training set are built by augmentation (Sect. 3.1.1). The goal of the trained model is therefore to assign a confidence value between 0 (no confidence) and 1 (maximum confidence) for each of these new MLAR sequences to belong to the c_+ class. Computing a confidence score for each individual pixel leads to a confidence map, from which exoplanet detection can be performed by choosing a confidence threshold.

3.2. Model adaptation: The NA-SODINN algorithm

In SODINN, the training set is built by mixing all MLAR sequences from the same class, generated on every annulus in the field of view. In the presence of different noise regimes, this way of proceeding can complicate the training of the model, as the statistics of an MLAR sequence generated in the speckle-dominated regime differ from a sequence of the same class generated in the background-dominated regime instead. In order to deal with this, we train an independent SODINN detection model per noise regime instead of a unique model for the full frame field of view. Thereby, each detection model is only trained with those MLAR sequences that contain statistical properties from the same (or similar) probabilistic distribution function. Therefore, our region of interest in the field of view is now smaller. This means that the number of pixels available to generate MLAR sequences is reduced, and therefore, that we are losing noise diversity in comparison with a model that is trained in the full frame. However, this loss of diversity comes with the benefit of better capturing the statistics of noise within a same noise regime, which improves the training.

3.2.1. Adding S/N curves to the network

In order to compensate for the noise diversity loss associated with the training on individual noise regimes, we attempt to reinforce the training by means of new handcrafted features. An interesting discriminator between the c_+ and c_- classes, which is also physically motivated, comes from their behaviour in terms of signal-to-noise ratio (S/N). The most accepted and used S/N definition in the HCI literature is from Mawet et al. (2014). It states that, given a $1\lambda/D$ wide annulus in a processed frame at distance r (in λ/D units) from the star, paved with $N = 2\pi r$ non-overlapping circular apertures (see Fig. 2), the S/N for one of these apertures is defined as

$$S/N = \frac{\bar{x}_t - \bar{x}_{N-1}}{\sigma_{N-1}\sqrt{1+\frac{1}{N-1}}}, \tag{2}$$

where $\bar{x}_t$ is the aperture flux photometry in the considered test aperture, $\bar{x}_{N-1}$ the average intensity over the remaining $N-1$ apertures in the annulus, and σ_{N-1} their standard deviation. In order to maximize the S/N, image processing detection algorithms need to be tuned through finding the optimal configuration of their parameters (see e.g. Dahlqvist et al. 2021b). Here, rather than optimizing the algorithm parameters, we use the fact that we can leverage the behaviour of the S/N versus some of the algorithm parameters in our deep learning approach. This is especially the case for the number of principal components used in the PSF subtraction. We define an S/N curve as the evolution of the S/N computed for a given circular aperture as a function of the number of principal components

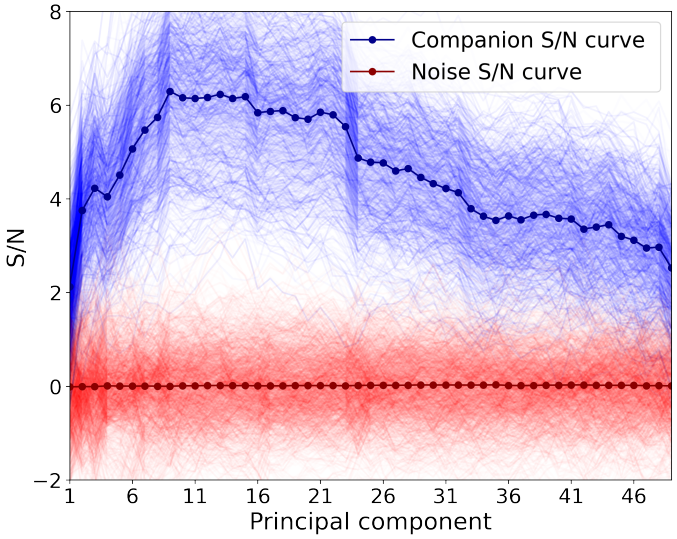


Fig. 7. S/N curves generated from the *sph*2 cube of processed frames at a $8\lambda/D$ distance from the star. Curves in blue contain the exoplanet signature and curves in red just residual noise. The flux of injections is randomly selected from a range that is between one and three times the level of noise. Dotted curves over populations show the mean of each class.

(Gomez Gonzalez et al. 2017). Figure 7 shows an example of 1000 S/N curves generated from the *sph*2 ADI sequence. We clearly see in Fig. 7 that, in the presence of an exoplanet signature (blue curves), the S/N curve first increases and then decreases, which leads to the appearance of a peak at a given number of principal components. This behaviour, capturing the competition between noise subtraction and signal self-subtraction, was already documented elsewhere (e.g. Gomez Gonzalez et al. 2017). The peak in the S/N curve indicates the number of principal components for which the contrast between the companion and the residual noise in the annulus is maximum. Hereafter, we denote as k the principal component at which this S/N peak is located.

For a given 1-FWHM circular aperture, the MLAR sequence (no matter the class) and the S/N curve are linked from a physical point of view. Actually, the evolution of the S/N as a function of the number of principal components can be readily extracted from intermediate products used in the production of the training dataset. Therefore, the information conveyed through the S/N curve is already partly contained in the MLAR patches. But while the MLAR sequence contains localized information on the signal and noise behaviour, the S/N curve conveys annulus-wise information, obtained through aperture photometry. Indeed, each aperture's S/N estimation depends on the noise in the rest of the annulus (Eq. (2)), so it also contains information that connects with other circular apertures at the same angular separation from the star. This dependency is not captured in MLAR sequences. S/N curves make this rich summary statistics directly available to the neural network to improve the neural network training. One complication in using S/N curves in the training relates to data augmentation, which is mandatory to build up a sufficiently large training dataset for SODINN. Because these augmentation operations modify the intensity and distribution of pixels in the MLAR sequence, there is no direct way to compute the associated S/N curve of an augmented MLAR sequence through Eq. (2). To deal with this, we make simplifying assumptions for each augmentation operation in SODINN: (i) image rotations do

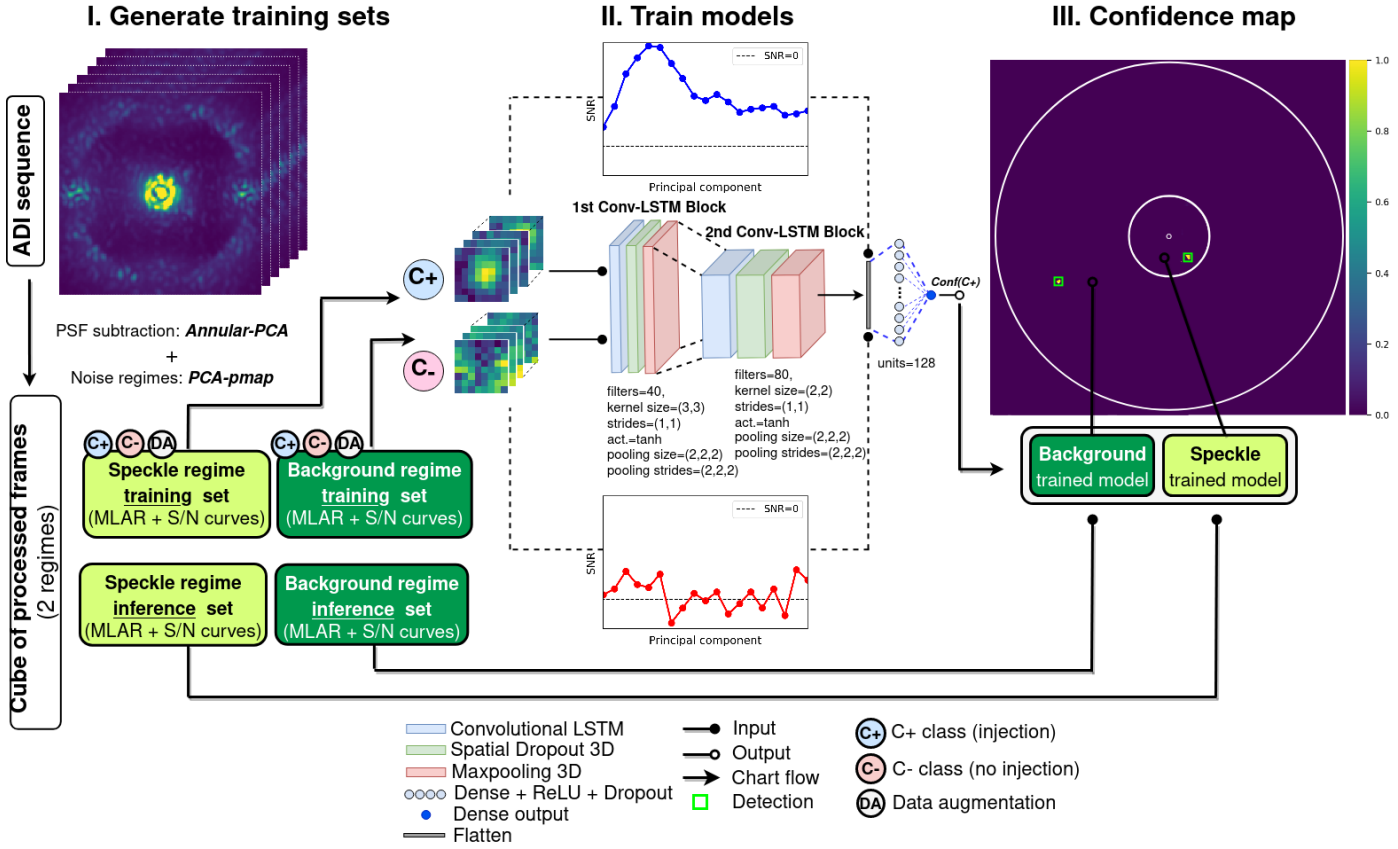


Fig. 8. Illustration of the three steps within the NA-SODINN algorithm working flow. *Left*: generation of the training set. NA-SODINN uses the annular-PCA algorithm to perform PSF subtraction and produce the cube of processed frames. Then, it detects residual noise regimes by applying the PCA-pmap technique to this cube, and builds both the training and inference datasets at each regime, which are composed of both MLAR samples and S/N curves. *Middle*: model training. NA-SODINN trains as many detection models as detected noise regimes using their respective training datasets (for the sake of simplicity, we have not duplicated the central deep neural network). This case contains two noise regimes, the speckle- and background-dominated noise regimes, so that two models are trained. *Right*: detection map. Finally, NA-SODINN uses each trained model to assign a confidence value to belong to the c_+ class to each pixel of the corresponding noise regime field of view.

not affect the S/N curve as the same pixels are kept in the final sequence, (ii) averaging two sequences can be approximated as averaging their S/N curves, and (iii) image shifts do not affect the S/N curve as long as the shift is sufficiently small.

By adding the noise regime approach and the S/N curves to SODINN, we are building a new detection algorithm. We refer to this novel framework, depicted in Fig. 8, as Noise-Adaptive SODINN, or NA-SODINN for short. As its predecessor, NA-SODINN is composed of the same three steps: producing the training set from an ADI sequence (Sect. 3.2.2), training a detection model with this training set, and applying the model to find companions in the same ADI sequence (Sect. 3.2.3).

3.2.2. Generation of the training set

NA-SODINN generates as many training sets as detected residual noise regimes. Each of these sets is composed of MLAR sequences and their corresponding S/N curves generated from the corresponding noise regime, including data augmentation.

Unlike SODINN, which makes use of the CEVR to define the appropriate range of principal components to generate the MLAR sequences (Gomez Gonzalez et al. 2018), the selection of the principal components for producing both MLAR sequences and S/N curves in NA-SODINN is instead determined through a novel metric derived from the PCA-pmap. For each rolling annulus, the PCA-pmap can be used to estimate the principal component k that maximizes the S/N for any planetary injection at any position within the annulus (see the peak on the blue curves of Fig. 7). The underlying motivation behind the identification of k is that MLAR sequences and their S/N curves can then be defined around this principal component, thus maximizing the gap between planetary and noise signals in the training set.

To identify k at a given angular separation and for a predefined S/N interval of injections, the PCA-pmap relies on two steps: (i) through the data-driven procedure of Appendix B, it pre-estimates the injection flux range that corresponds to the selected S/N range; (ii) once this flux range is estimated, it is used to randomly select fluxes within the range to inject many fake companions, within the annulus at random coordinates, and retrieve their S/N curves (e.g. Fig. 7). The k can finally be estimated by averaging all these S/N curves. Here, we select the injected companion fluxes to produce an S/N ranging between 1 and 3 in the final PCA-processed map obtained with one single principal component, which was experimentally found to be appropriate for the NA-SODINN training as it generally produces companions close to the detection limit for a larger number of PCs. We indicate the k obtained for this S/N range as white circles in Figs. 4a and 5a. By comparing k with the principal components where the 90% CEVR is reached in PCA-pmaps for both *sph*2 and *nrc*3 ADI sequences (Figs. 4a and 5a), we observe that at some angular separations, k is not well captured by the

CEVR metric. This suggests that the use of CEVR as a figure of merit for choosing the principal components is not always optimal. Therefore, while more training data is generally beneficial, the range of PCs around the value of k can be chosen differently each time NA-SODINN is employed. A range between 15 and 30 PCs is generally optimal.

3.2.3. Training and inference

NA-SODINN trains an independent detection model for each regime by using its corresponding training set. For each MLAR sequence in the training set, the feature maps created through convolutional blocks are now concatenated with their respective S/N curves after the flattened layer (Fig. 8). NA-SODINN generally reaches a ∼99.9% validation accuracy with 5–8 epochs. In the last step, NA-SODINN makes inferences for each individual noise regime. It applies the trained model of each regime to infer its corresponding confidence map of the same regime (Fig. 8). Finally, NA-SODINN builds the final confidence detection map by joining all confidence regime maps inferred with each detection model. Thus, our NA-SODINN algorithm is conceived to keep the main characteristics of the pioneering SODINN algorithm (Gomez Gonzalez et al. 2018), such as its architecture, and adapt its optimization process to our local noise approach.

4. Model evaluation

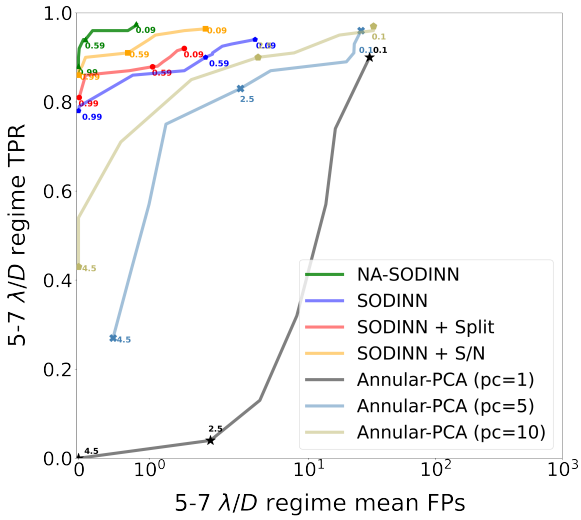
Now that NA-SODINN has been introduced, we aim to thoroughly evaluate its detection ability. In the first part of this section, we explain the evaluation strategy and benchmark NA-SODINN with respect to its predecessor SODINN using the same *sph2* and *nrc3* ADI sequences. Then, in the second part, we apply NA-SODINN to the first phase of the EIDC (Cantalloube et al. 2020), providing confidence maps for each ADI sequence in the data challenge and running the same statistical analysis to compare the NA-SODINN performance with the rest of the HCI algorithms.

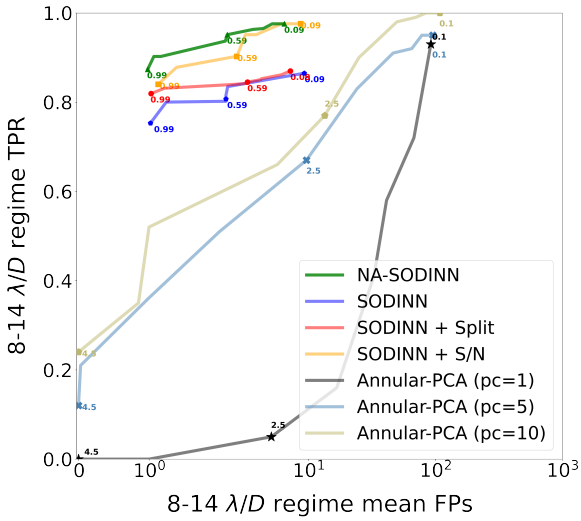
4.1. Performance assessment

The evaluation of HCI detection algorithms consists of minimizing the false positive rate (FPR) while maximizing the true positive rate (TPR) at different detection thresholds applied in the final detection map. This information is summarized by a curve in the receiver operating characteristics (ROC) space, where each point in the curve captures both metrics at a given threshold value (Gomez Gonzalez et al. 2018; Dahlqvist et al. 2020). In order to produce ROC curves for various versions of SODINN applied to a given ADI sequence D, we first build the evaluation set $\mathcal{D}_{eval} = \{D_1, D_2, D_3, \ldots, D_s\}$ containing s synthetic datasets D_i, where each synthetic dataset is a copy of D with one fake companion injection per noise regime. Here, we limit the number of injected companions per noise regime to one at a time to avoid any risk of cross-talk between companions in the detection algorithms themselves (e.g. because multiple companions can affect the PCA), or in their evaluation (e.g. if they get too close and merge in terms of confidence patch). The coordinates of these injections are randomly selected within the considered noise regime boundaries, and their fluxes are randomly set within a pre-defined range of fluxes that correspond to an S/N range between 0.5 and 2 in the processed frame. This pre-defined range of fluxes is estimated through the same data-driven strategy explained in Appendix B and illustrated in Fig. B.1. Hence, each algorithm provides s final detection maps, from which true positives (TPs), false positives (FPs), true negatives (TNs) and false negatives (FNs) indicators are computed across the whole noise regime field of view at different detection thresholds. Then, all these indicators are averaged, and the corresponding ROC curve for the considered noise regime is produced. Instead of using the FPR metric as in standard ROC curves, here we used the mean number FPs within the whole field of view, which is more representative of the HCI detection task and facilitates the interpretation of our performance simulations.

We perform the proposed ROC curve analysis on both *sph*2 and *nrc*3 ADI sequences, with $s = 100$ for each. For this assessment, a detection is defined as a blob in the final detection map with at least one pixel above the threshold inside a circular aperture of diameter equal to the FWHM centred at the position of each injection. With the aim of benchmarking NA-SODINN, we include in this evaluation the annular-PCA algorithm (Absil et al. 2013) as implemented in the `VIP` Python package (Gomez Gonzalez et al. 2017; Christiaens et al. 2023), the SODINN framework by Gomez Gonzalez et al. (2018), and two hybrid detection models. These hybrid models are modifications of SODINN to include only one of the two additional features introduced in NA-SODINN: the adaptation to noise regimes, or the addition of S/N curves in the training. Hereafter, we refer to them respectively as SODINN+Split and SODINN+S/N. In the same spirit as an ablation study, these two hybrid models are included in our evaluation in order to provide information about the added value of each approach separately for the task of detection. It is worth mentioning that instead of retraining all considered SODINN-based models every time a different fake companion is injected into each evaluation set, we train them once per ADI sequence. While retraining would be more accurate, as the presence of an injected fake companion could slightly perturb the c_- class, we assume that our augmentation strategy (Sect. 3.1.1) mitigates this perturbation and does not significantly impact the training process and the model's performance. Moreover, using the same model to detect all fake companion injections in a single ADI sequence saves computation time.

An important aspect to consider when comparing algorithms in ROC space is to optimally choose their model parameters. In the case of annular-PCA, we use one, five, and ten principal components for each annulus as a good compromise to analyse its performance. For the various versions of SODINN, we need to define two main parameters: the list of principal components $\mathcal{PC} = (\mathrm{pc}_1, \mathrm{pc}_2, \ldots, \mathrm{pc}_m)$ that are used to produce each sample in both the MLAR sequence and S/N curve, and the level of injected fluxes used for making c_+ class samples (see Sect. 3.1). For SODINN, we used the criterion based on the CEVR, as proposed by Gomez Gonzalez et al. (2018), to define the $\mathcal{PC}$ list. For NA-SODINN and the hybrid models, we instead rely on the novel PCA-pmaps technique, and we choose a list of $m = 15$ principal components centred around k (Sect. 3.2.2). Regarding the injected fake companion fluxes, we choose for all SODINN-based models a range of fluxes that correspond to an S/N between one and three in the PCA-processed frame with one principal component (Appendix B). This range of fluxes does not generally lead to class overlapping, where c_+ and c_- class samples would look too similar. However, in order to avoid FPs in the final detection map, the user may consider higher flux ranges. Finally, to build the ROC curve, we consider a list of S/N thresholds ranging from 0.1 to 4.5 in steps of 0.5 for annular PCA, while for the SODINN-based models, we use a list of confidence thresholds from 0.09 to 0.99 in steps of 0.1. All SODINN-based





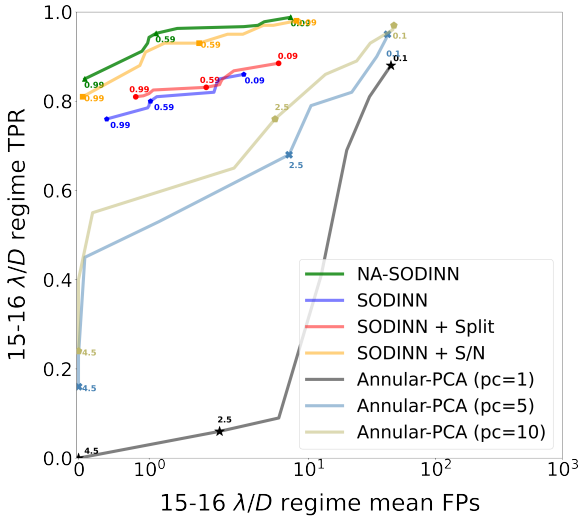
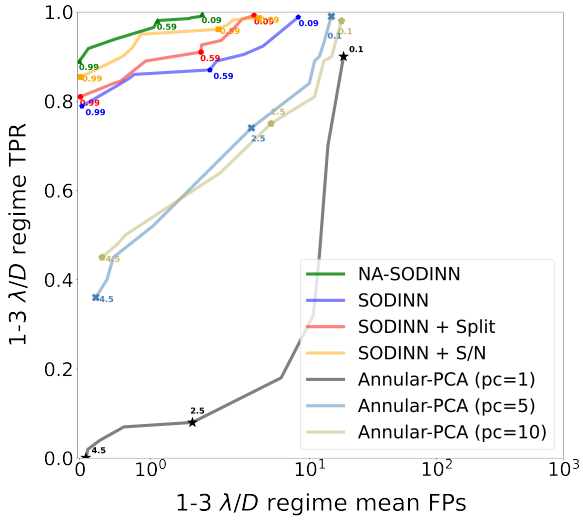


Fig. 9. ROC analysis per noise regime for the *sph2* dataset showing the performance of SODINN, NA-SODINN, annular-PCA, and hybrid SODINN models. The values plotted alongside each curve highlight some of the selected thresholds.



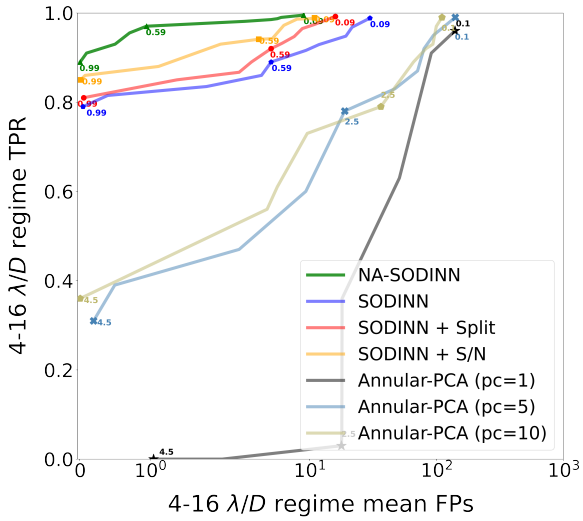


Fig. 10. Same as Fig. 9, but for the *nrc3* dataset.

models are trained on balanced training sets containing around 10^5 samples for each class using an NVIDIA GeForce RTX 3070 graphics processing unit (GPU).

Figures 9 and 10 display a series of ROC spaces – one for each detected noise regime –, respectively for the *sph2* and *nrc3* ADI sequences. For the sake of simplicity, we do not consider the detected regime comprised between 17 and 19 λ/D in *sph2* (Fig. 4) for this analysis. Each of these ROC spaces displays one ROC curve per algorithm, which informs about its detection performance on that specific noise regime for different thresholds. We observe from both figures that NA-SODINN outperforms its predecessor, the hybrid models, and the annular-PCA technique for each noise regime. This behaviour is further illustrated in Appendix C, with Figs. C.1, C.2 and C.3 for the case of *sph2*, and Figs. C.4, C.5 for *nrc3*, where the confidence maps from each algorithm are compared at different threshold levels. Regarding hybrid models, we generally observe that they land between the SODINN and NA-SODINN detection performance, with SODINN+S/N generally being the best hybrid model. It can also be observed that annular-PCA with PC=5 and PC=10 perform better than with PC=1 for all regimes. We associate this behaviour to the fact that for PC=5 and PC=10, we are closer

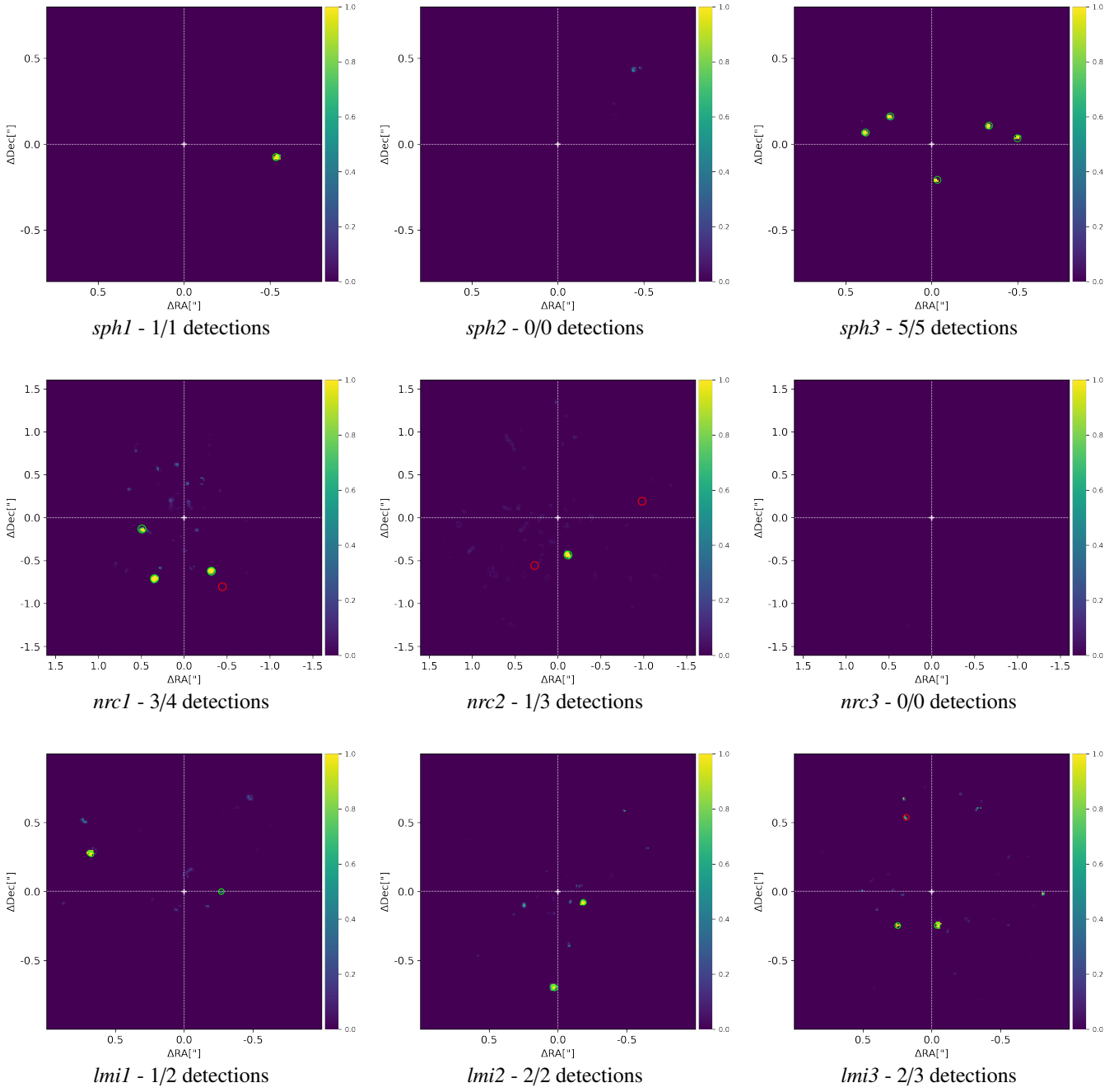


Fig. 11. NA-SODINN confidence maps obtained on the whole set of EIDC ADI sequences (Table A.1). For the submitted confidence threshold $\tau = 0.90$, we highlight with green circles the correct detection of injected companions (true positives), and with red circles the non-detection of injected companions (false negatives). The circles have a FWHM diameter. No false positive is reported in our maps, as all the remaining non-circled peaks in the confidence maps are below the threshold.

to the principal component k where the S/N is maximized, and therefore, the star-planet contrast is improved.

Based on these results and our experiments, we observe a general trend for both approaches separately. While splitting the field of view in noise regimes tends to reduce the number of false positives, especially when residual speckle noise is significant, adding an S/N curve for each MLAR sequence tends to enhance the algorithm's sensitivity to detect signals. These findings imply that both techniques, when combined in the neural network, considerably improve the SODINN detection performance.

4.2. NA-SODINN in EIDC

By design, the Exoplanet Imaging Data Challenge (EIDC, Cantalloube et al. 2020) can be used as a laboratory to compare and evaluate new detection algorithms against other state-of-the-art HCI detection algorithms. For instance, Dahlqvist et al. (2021a) used the EIDC to highlight the improvement of the automated version of their RSM algorithm. Here, we use the first sub-challenge of the EIDC to generalize the ROC analysis presented above, and evaluate how NA-SODINN performs with respect to the state-of-the-art HCI algorithms that entered the data challenge. Besides the *sph*2 and *nrc*3 datasets used so

far, the first EIDC sub-challenge includes seven additional ADI sequences in which a total of 20 planetary signals with different contrasts and position coordinates were injected. Two of these seven ADI sequences are from the SPHERE instrument (Beuzit et al. 2019), identified as *sph1* and *sph3*, two more from the NIRC-2 instrument (Serabyn et al. 2017), identified as *nrc1* and *nrc3*, and the remaining three from the LMIRCam instrument (Skrutskie et al. 2010), with *lmr1*, *lmr2* and *lmr3* ID names. For each of these nine datasets, EIDC provides a pre-processed temporal cube of images, the parallactic angles variation corrected from true north, a non-coronagraphic PSF of the instrument, and the pixel-scale of the detector. Each algorithm entering the EIDC had to provide a detection map for each ADI sequence. The following standard metrics are then used to assess the detection performance of each submitted detection map:

- True Positive Rate: TPR = $\frac{TP}{TP+FN}$,
- False Positive Rate: FPR = $\frac{FP}{FP+TN}$,
- False Discovery Rate: FDR = $\frac{FP}{FP+TP}$,
- F1-score: F1 = $\frac{2 \cdot TP}{2 \cdot TP+FP+FN}$.

We apply our NA-SODINN framework to the EIDC, and as in the ROC analysis, we use PCA-pmaps as a tool for both estimating residual noise regimes and choosing the list of principal components $\mathcal{PC}$ at each angular separation. For the injection flux ranges, we use an S/N range between one and four times the level of noise in the processed frame. Each model is trained with balanced training sets that contain around 10^5 samples per class. Because all three LMIRCam cubes contain more than 3,000 frames (Table A.1), we decided to reduce this number to around 250–300 frames to limit the computational time. To do that, we average a certain number of consecutive frames along the time axis in the sequence. Figure 11 shows a grid of all the resulting NA-SODINN confidence maps for the EIDC ADI sequences where we observe, by visual inspection, that NA-SODINN finds most of the injected fake companions, while producing only faint false positives that all fall below our default detection threshold $\tau = 0.9$. In order to quantify this information, we follow the same approach as in Cantalloube et al. (2020) by considering the area under the curve (AUC) for the TPR, FPR, and FDR as a function of the threshold, which allows mitigating the arbitrariness of the threshold selection by considering their evolution for a predefined range. The $\mathrm{AUC_{TPR}}$ should be as close as possible to one, and the $\mathrm{AUC_{FPR}}$ and $\mathrm{AUC_{FDR}}$ as close as possible to zero. The F1-score ranges between zero and one, where one corresponds to a perfect algorithm, and is computed only on a single threshold τ_{sub} that is chosen by the participant.

Figure D.1 shows the result of this analysis for all NA-SODINN confidence maps of Fig. 11, in which all TPR, FPR, and FDR metrics (and their respective AUCs) are computed for different confidence threshold values ranging from zero to one. Here, we mainly see that the $\mathrm{AUC_{FDR}}$ is generally higher along the range of thresholds for NIRC-2 and LMIRCam than for SPHERE datasets, the $\mathrm{AUC_{FPR}}$ is close to zero for all datasets, and the $\mathrm{AUC_{TPR}}$ is almost perfect for SPHERE datasets. To compute the F1-score, we choose a $\tau_{\mathrm{sub}} = 0.9$ confidence threshold. From our test with NA-SODINN, we consider this value as the minimum confidence threshold for which one can rely on the significance of detections, maximizing TPs while minimizing FPs. Thus, any pixel signal above this τ_{sub} on each confidence map of Fig. 11 is considered as a detection for the computation of the F1-score. Finally, through the $\mathrm{AUC_{TPR}}$, $\mathrm{AUC_{FDR}}$ and F1-score metrics obtained with the NA-SODINN algorithm, we are able to update the general EIDC leaderboard (Cantalloube et al. 2020). Figure 12 shows how NA-SODINN ranks compared to the

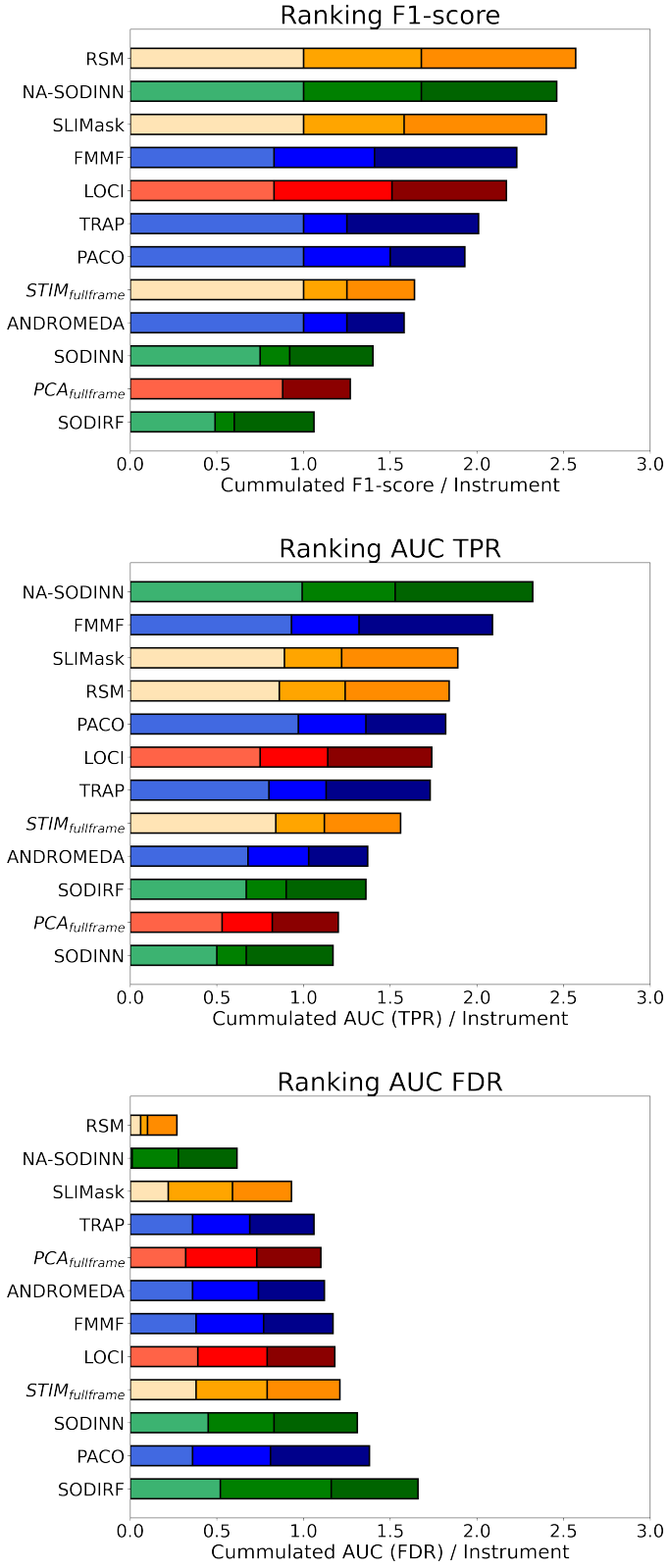


Fig. 12. Updated EIDC leaderboard after the NA-SODINN submision. Ranking based on the $F1$-score (on *top*), the AUC of the TPR (in the *middle*) and the AUC of the FDR (on *bottom*). Colours refer to HCI detection algorithm families: PSF-based subtraction techniques providing residual maps (red) or detection maps (orange), inverse problems (blue) and supervised machine learning (green). The light, medium and dark tonalities correspond to SPHERE, NIRC-2, and LMIRCam datasets, respectively.

algorithms originally submitted to the EIDC, for each considered metric. We clearly observe that NA-SODINN ranks at the top, or close to the top, for each of the EIDC metrics, with results generally on par with the RSM algorithm by Dahlqvist et al. (2020). In particular, NA-SODINN provides the highest area under the true positive curve, while preserving a low false discovery rate.

5. Conclusions

In this paper, we explore the possibility of enhancing exoplanet detection in the field of HCI by training a supervised classification model that takes into account the noise structure in the PCA-processed frame. SODINN (Gomez Gonzalez et al. 2018), a pioneering deep-learning detection algorithm in HCI, has been adapted to learn from different noise regimes in the processed frame and local discriminators between the exoplanet and noise, such as S/N curves. With these two approaches working in synergy, we built a new detection algorithm, NA-SODINN. Although our findings related to the spatial structure of noise distributions are showcased by adapting the SODINN detection framework, we believe that other algorithms dealing with processed frames could be adapted similarly.

The NA-SODINN detection capabilities were tested through two distinct analyses. First, we performed a performance assessment based on ROC curves using two ADI sequences provided by the VLT/SPHERE and Keck/NIRC-2 instruments. Here, NA-SODINN is evaluated with respect to annular-PCA, the original SODINN, and two SODINN-based hybrid models that use only one of the two proposed approaches, that is, the noise regime splitting or the S/N curves' addition. We found that hybrid models improve the detection performance of SODINN in all noise regimes, which demonstrates the interest of the local noise approaches considered in this paper. Moreover, we found that NA-SODINN reaches an even higher detection performance, especially in the speckle noise regime, by combining both approaches in the same framework. Then, in order to benchmark NA-SODINN against other state-of-the-art HCI algorithms, we applied NA-SODINN to the first phase of EIDC (Cantalloube et al. 2020), a community-wide effort meant to offer a platform for a fair and common comparison of exoplanetary detection algorithms. In this analysis, we observed that NA-SODINN is ranked at the top (first or second position) of the challenge leaderboard for all considered evaluation metrics, providing in particular the highest true positive rate among all entries, while still keeping a low false discovery rate.

We identified some limitations that could be addressed in future work to improve the effectiveness and practicality of our NA-SODINN method. While the algorithm currently performs well in noise regimes over PCA-processed frames, it relies on previous noise analyses to define these regime boundaries, limiting its independence. Future avenues would include modifying the network architecture to enable the identification of noise regimes during training, which could enhance the detection performance. Another limitation of our approach is the challenge of setting an appropriate detection threshold in the final detection map. This is typically based on the presence of obvious false positives, which may affect the application of NA-SODINN in certain contexts. However, this limitation can be mitigated by using dedicated metrics such as ROC space to assess detection performance. We also note that NA-SODINN and its predecessor rely on data augmentation techniques to generate a diverse training set. To supplement these techniques, we suggest exploring generative neural networks to train more robust supervised models that can generalize better. Lastly, extending the application of NA-SODINN to work on other observing strategies and detect extended sources such as protoplanetary disks would be a valuable avenue to increase the flexibility of the algorithm.

The NA-SODINN framework represents a significant step forward in the search for new and unconfirmed worlds in individual datasets and large surveys using ADI-based techniques. This framework offers greater accuracy in identifying exoplanets across all angular separations, making it particularly well suited for improving our understanding of the demographics of directly imaged exoplanets.

Acknowledgements. The authors would like to thank the python open-source scientific community, and in particular the developers of the Keras deep learning library (Abadi et al. 2015) and the VIP high-contrast imaging package (Gomez Gonzalez et al. 2017; Christiaens et al. 2023). The authors acknowledge stimulating discussions with Faustine Cantalloube, Rakesh Nath, Markus Bonse, and Emily O. Garvin, as well as the whole Exoplanet Imaging Data Challenge team. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement no. 819155), and from the Wallonia-Brussels Federation (grant for Concerted Research Actions).

References

Abadi, M., Agarwal, A., Barham, P., et al. 2015, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, software available from tensorflow.org
Absil, O., Milli, J., Mawet, D., et al. 2013, A&A, 559, L12
Ahmad, F., & Khan, R. A. 2015, Pak. J. Stat. Oper. Res., 11, 331
Amara, A., & Quanz, S. P. 2012, MNRAS, 427, 948
Anderson, T. W., & Darling, D. A. 1952, Ann. Math. Stat., 23, 193
Beuzit, J.-L., Vigan, A., Mouillet, D., et al. 2019, A&A, 631, A155
Bohn, A. J., Ginski, C., Kenworthy, M. A., et al. 2021, A&A, 648, A73
Bonse, M., Garvin, E., Gebhard, T., et al. 2022, Bull. Am. Astron. Soc., 54, 5
Boureau, Y.-L., Ponce, J., & LeCun, Y. 2010, in International Conference on Machine Learning (ICML), Haifa, Israel, 111
Bulmer, M. G. 1979, Principles of Statistics (Mineola, New York, USA: Dover Publications)
Cantalloube, F., Mouillet, D., Mugnier, L. M., et al. 2015, A&A, 582, A89
Cantalloube, F., Dohlen, K., Milli, J., Brandner, W., & Vigan, A. 2019, The Messenger, 176, 25
Cantalloube, F., Gomez-Gonzalez, C., Absil, O., et al. 2020, Proc. SPIE, 11448, 114485A
Chauvin, G., Desidera, S., Lagrange, A.-M., et al. 2017, A&A, 605, L9
Christiaens, V., Gonzalez, C. A. G., Farkas, R., et al. 2023, J. Open Source Softw., 8, 4774
D'Agostino, R., & Pearson, E. S. 1973, Biometrika, 60, 613
Dahlqvist, C.-H., Cantalloube, F., & Absil, O. 2020, A&A, 633, A95
Dahlqvist, C.-H., Cantalloube, F., & Absil, O. 2021a, A&A, 656, A54
Dahlqvist, C.-H., Louppe, G., & Absil, O. 2021b, A&A, 646, A49
Flasseur, O., Denis, L., Thiébaut, É., & Langlois, M. 2018, A&A, 618, A9
Flasseur, O., Bodrito, T., Mairal, J., et al. 2023, MNRAS, 527, 1534
Gebhard, T. D., Bonse, M. J., Quanz, S. P., & Schölkopf, B. 2022, A&A, 666, A9
Gneiting, T. 1997, J. Stat. Comput. Simul., 59, 375
Goebel, S. B., Guyon, O., Hall, D. N. B., Jovanovic, N., & Atkinson, D. E. 2016, Proc. SPIE, 9909, 417
Gomez Gonzalez, C., Absil, O., Absil, P.-A., et al. 2016, A&A, 589, A54
Gomez Gonzalez, C., Wertz, O., Absil, O., et al. 2017, AJ, 154, 7
Gomez Gonzalez, C., Absil, O., & Van Droogenbroeck, M. 2018, A&A, 613, A71
Halko, N., Martinsson, P.-G., Shkolnisky, & Tygert, M. 2011, SIAM J. Sci. Comput., 33, 2580
Hinkley, S., Oppenheimer, B. R., Soummer, R., et al. 2007, ApJ, 654, 633
Jensen-Clem, R., Mawet, D., Gomez Gonzalez, C. A., et al. 2017, AJ, 155, 19
Keppler, M., Benisty, M., Müller, A., et al. 2018, A&A, 617, A44
Lafreniere, D., Marois, C., Doyon, R., Nadeau, D., & Artigau, E. 2007, ApJ, 660, 770
Lilliefors, H. W. 1967, J. Am. Stat. Assoc., 62, 399
Lozi, J., Guyon, O., Jovanovic, N., et al. 2018, Proc. SPIE, 10703, 1070359
Males, J. R., Fitzgerald, M. P., Belikov, R., & Guyon, O. 2021, PASP, 133, 104504
Marmolejo-Ramos, F., & González-Burgos, J. 2013, Methodology, 9, 137

Marois, C., Lafreniere, D., Doyon, R., Macintosh, B., & Nadeau, D. 2006, ApJ, 641, 556
Marois, C., Lafreniere, D., Macintosh, B., & Doyon, R. 2008a, ApJ, 673, 647
Marois, C., Macintosh, B., Barman, T., et al. 2008b, Science, 322, 1348
Marois, C., Zuckerman, B., Konopacky, Q. M., Macintosh, B., & Barman, T. 2010, Nature, 468, 1080
Marois, C., Correia, C., Galicher, R., et al. 2014, Proc. SPIE, 9148, 91480U
Mawet, D., Serabyn, E., Liewer, K., et al. 2009, ApJ, 709, 53
Mawet, D., Milli, J., Wahhaj, Z., et al. 2014, ApJ, 792, 97
Nair, V., & Hinton, G. 2010, in International Conference on Machine Learning (ICML), Haifa, Israël, 807
Pairet, B., Cantalloube, F., Gomez Gonzalez, C. A., Absil, O., & Jacques, L. 2019, MNRAS, 487, 2262
Patrício, M., Ferreira, F., Oliveiros, B., & Caramelo, F. 2017, Commun. Stat. Simul. Comput., 46, 7535
Rameau, J., Chauvin, G., Lagrange, A.-M., et al. 2013, ApJ, 772, L15
Ren, B., Pueyo, L., Zhu, G. B., Debes, J., & Duchêne, G. 2018, ApJ, 852, 1
Ruffio, J.-B., Macintosh, B., Wang, J. J., et al. 2017, ApJ, 842, 14
Samland, M., Bouwman, J., Hogg, D. W., et al. 2021, A&A, 646, A24
Schölkopf, B., Hogg, D. W., Wang, D., et al. 2016, PNAS, 113, 7391
Serabyn, E., Huby, E., Matthews, K., et al. 2017, AJ, 153, 43
Shapiro, S. S., & Wilk, M. B. 1965, Biometrika, 52, 591
Shi, X., Chen, Z., Wang, H., et al. 2015, in Advances in Neural Information Processing Systems, 1 (NeurIPS), 802
Skrutskie, M. F., Jones, T., Hinz, P., et al. 2010, Proc. SPIE, 7735, 77353H
Snik, F., Absil, O., Baudoz, P., et al. 2018, Proc. SPIE, 10706, 107062L
Soummer, R. 2005, ApJ, 618, L161
Soummer, R., Ferrari, A., Aime, C., & Jolissaint, L. 2007, ApJ, 669, 642
Soummer, R., Pueyo, L., & Larkin, J. 2012, ApJ, 755, L28
Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. 2014, J. Mach. Learn. Res., 15, 1929
Uhm, T., & Yi, S. 2021, Commun. Stat. Simul. Comput., 1
Vovk, V., & Wang, R. 2020, Biometrika, 107, 791
Wagner, K., Apai, D., Kasper, M., et al. 2016, Science, 353, 673
Wahhaj, Z., Cieza, L. A., Mawet, D., et al. 2015, A&A, 581, A24
Wijekularathna, D. K., Manage, A. B. W., & Scariano, S. M. 2019, Commun. Stat. Simul. Comput., 51, 757
Yap, B. W., & Sim, C. H. 2011, J. Stat. Comput. Simul., 81, 2141
Yip, K. H., Nikolaou, N., Coronica, P., et al. 2020, in Lecture Notes in Computer Science, 11908, Joint European Conference on Machine Learning and Knowledge Discovery in Databases (Springer International Publishing), 322

Appendix A:EIDC datasets

Table A.1: Features of the nine ADI sequences from EIDC: The number of frames in the sequence (N_t), the frame size ($N_{\rm img}$), the wavelength (λ_{obs}), and the field rotation (Δ_{rot}).

ID	Telescope/Instr.	FWHM [px]	N_t	$N_{\rm img}$ [px×px]	λ_{obs} [μm]	Δ_{rot} [°]	Inj.
sph1	VLT/SPHERE	4	252	160×160	1.625 ± 0.29	40.3	1
sph2	VLT/SPHERE	4	80	160×160	1.593 ± 0.052	31.5	0
sph3	VLT/SPHERE	4	228	160×160	1.593 ± 0.052	80.5	5
nrc1	Keck/NIRC-2	9	29	321×321	3.776 ± 0.70	53.0	3
nrc2	Keck/NIRC-2	9	40	321×321	3.776 ± 0.70	37.3	4
nrc3	Keck/NIRC-2	9	50	321×321	3.776 ± 0.70	166.9	0
lmr1	LBT/LMIRCAM	5	4838	200×200	3.780 ± 0.10	153.4	2
lmr2	LBT/LMIRCAM	4	3219	200×200	3.780 ± 0.10	60.6	2
lmr3	LBT/LMIRCAM	4	4620	200×200	3.780 ± 0.10	91.0	3

Appendix B: Injection fluxes estimation

In HCI, a planetary injection is defined as the process of pasting the AO-corrected instrumental PSF (centred, cropped, and normalized) to every frame in the image sequence at specific coordinates (r, θ) following field rotation. To control the flux of this injection, the standard procedure is to multiply the normalized PSF by a flux scale factor α. Estimating an injection flux range that corresponds to a given S/N range in the post-processed frame implies estimating its respective flux scale factor range $\alpha_R = [\alpha_{min}, \alpha_{max}]$. Given a desired S/N range and an angular separation, α_{min} and α_{max} are estimated through the following data-driven procedure:

1. inject a companion in the raw image sequence at random coordinates (r, θ) within the annuli and with a random scale factor α;
2. compute the ADI-PCA processed frame for this synthetic image sequence using one single principal component in the PCA approximation of the speckle field;
3. apply Eq. 2 on the processed frame at the injection coordinates (r, θ), retrieving the companion S/N value;
4. repeat 1-3 steps $N_{\rm inj}$ times;
5. plot all S/N values retrieved from all N_{inj} injections of step 4 as a function of their corresponding scale factor;
6. linearly fit the data plotted in step 5, and define α_{min} and α_{max} as the intersection between the linear fit and the corresponding S/N range boundaries.

This process is repeated for each angular separation in the field of view in such a way that a different flux scale factor range α_R is estimated for each annulus. Figure B.1 illustrates this data-driven procedure for the case of the *sph2* dataset, showing the plots of step 5 for different annuli, each with $N_{\rm inj} = 3000$ injections (step 4). From Fig. B.1, we observe a general trend: the estimated scale factor range decreases as the angular distance increases. This observation aligns with our expectations, given that the spatial component of speckle noise intensity displays a radial dependency in the raw HCI data, a characteristic that persists even after the ADI processing. However, for this *sph2* dataset, a notable departure from this trend occurs between $15 - 16\lambda/D$ separations. In this specific interval, we observe in Fig. B.1 an anomalous increase in the estimated scale factors instead. We associate this behaviour with the fact that, at these angular separations for *sph2*, speckle dominates over background noise, as concluded in Sect. 2.

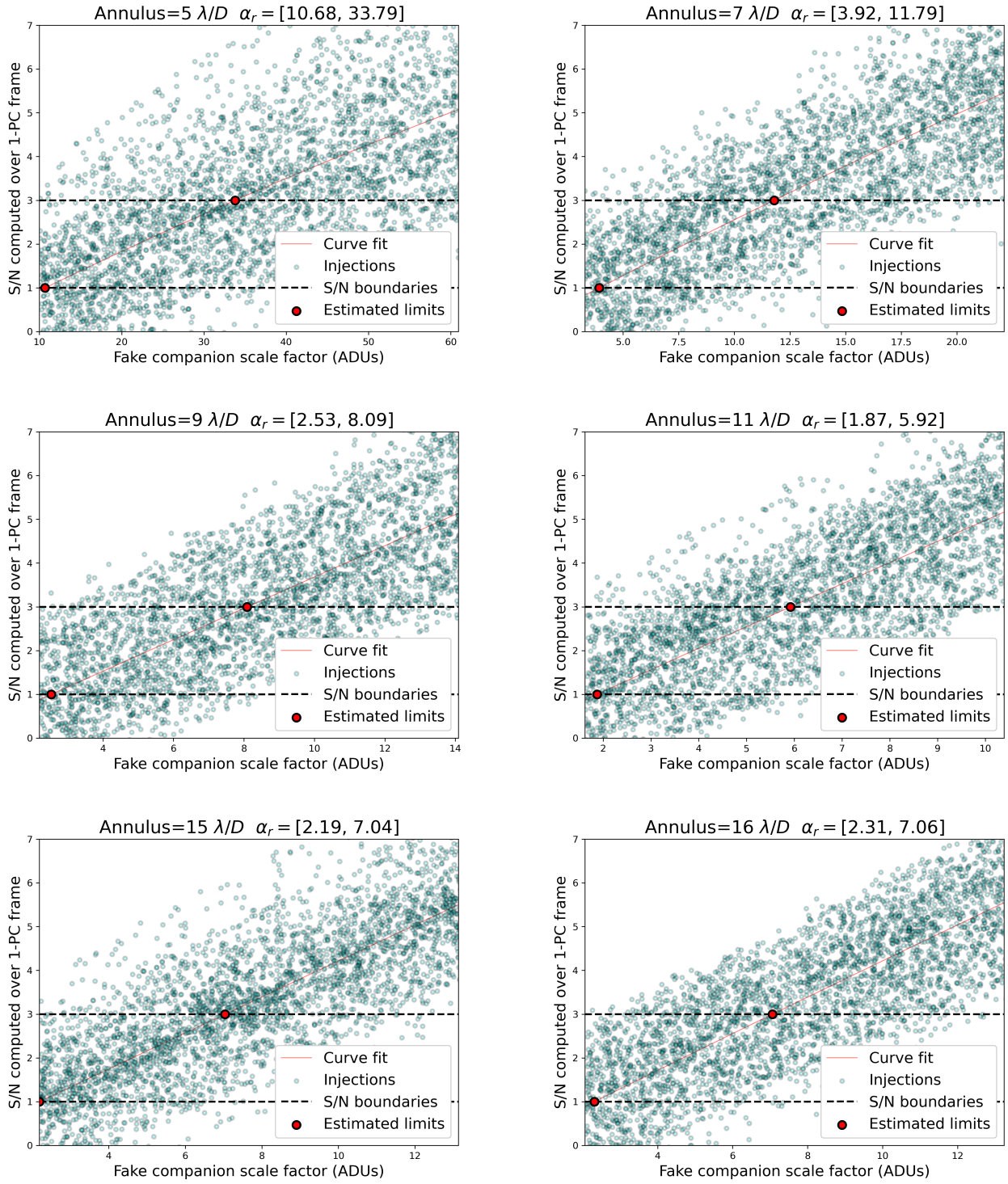


Fig. B.1: Example of the injection flux estimation method for the case of the *sph2* sequence. Each subplot refers to a different angular separation, and shows the S/N of an injection (y axis), retrieved from the PCA post-processed frame with one principal component, as a function of its scale factor (x axis). Each point in cyan on subplots thus represents a fake companion, which has been injected in the ADI sequence at random coordinates within the corresponding annulus and with a random scale factor. The thin red line is the curve fit of all injections, and dashed horizontal curves in black delimit the chosen S/N range, which is from one to three in this case. The two red dots show the intersection between the curve fit and the S/N limits (step 6), which therefore define the range of the scale factor corresponding to the chosen S/N limits.

Appendix C:Detection maps for *sph2* and *nrc3* datasets

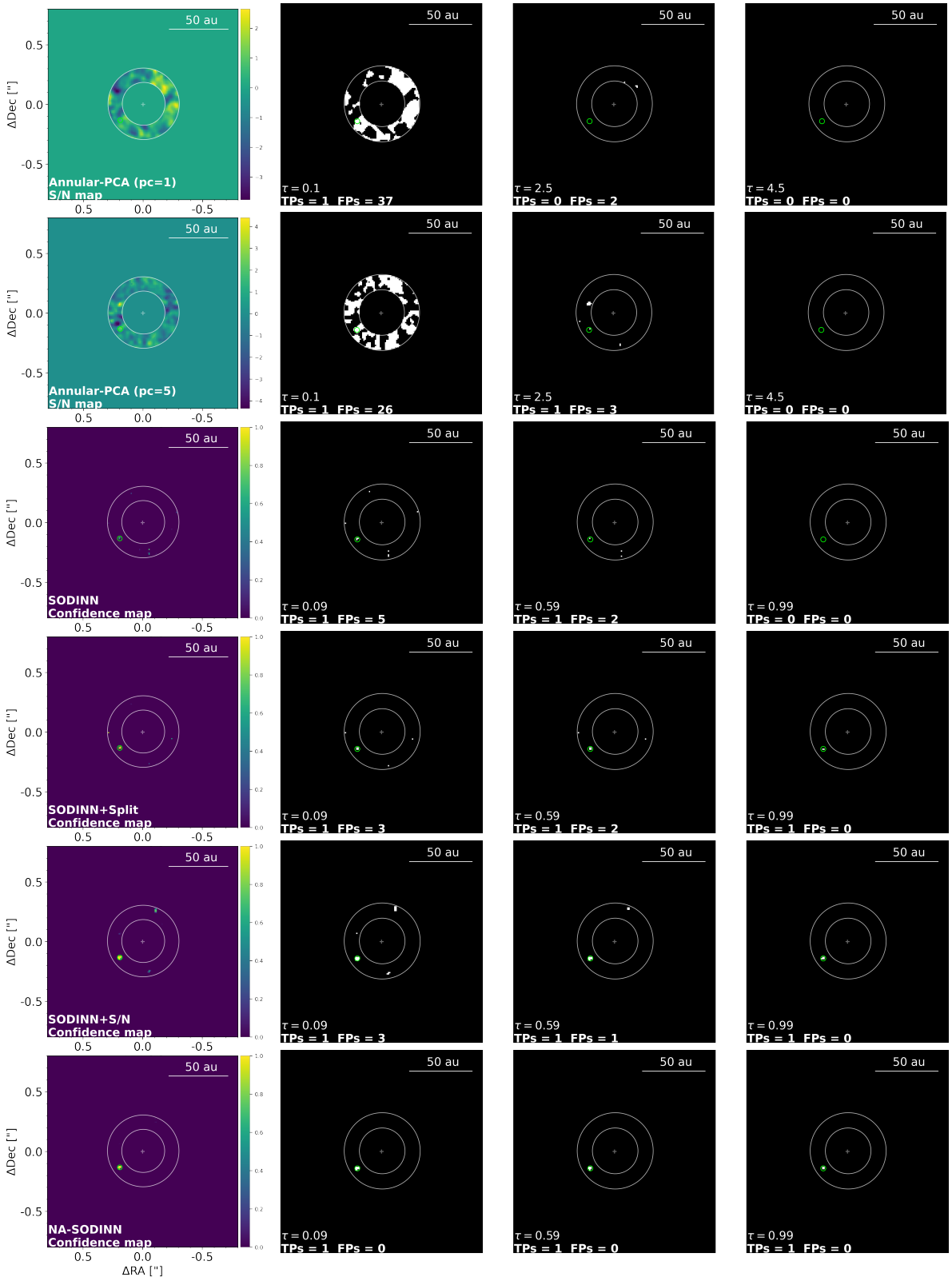


Fig. C.1: Evaluation example of Annular-PCA and all SODINN-based algorithms over the **5-7** λ/D regime of **sph2**, where a fake companion has been injected with **S/N=0.75** (computed in the PCA-processed frame using the first principal component). Each row corresponds to a different algorithm, where its detection map is on left column, and its three thresholds (binary maps) are on the right. The threshold τ, TPs and FPs are highlighted over each binary map. White concentric circles indicate the regime's boundaries. Other noise regimes are masked. Small green circles indicate the position of the injection.

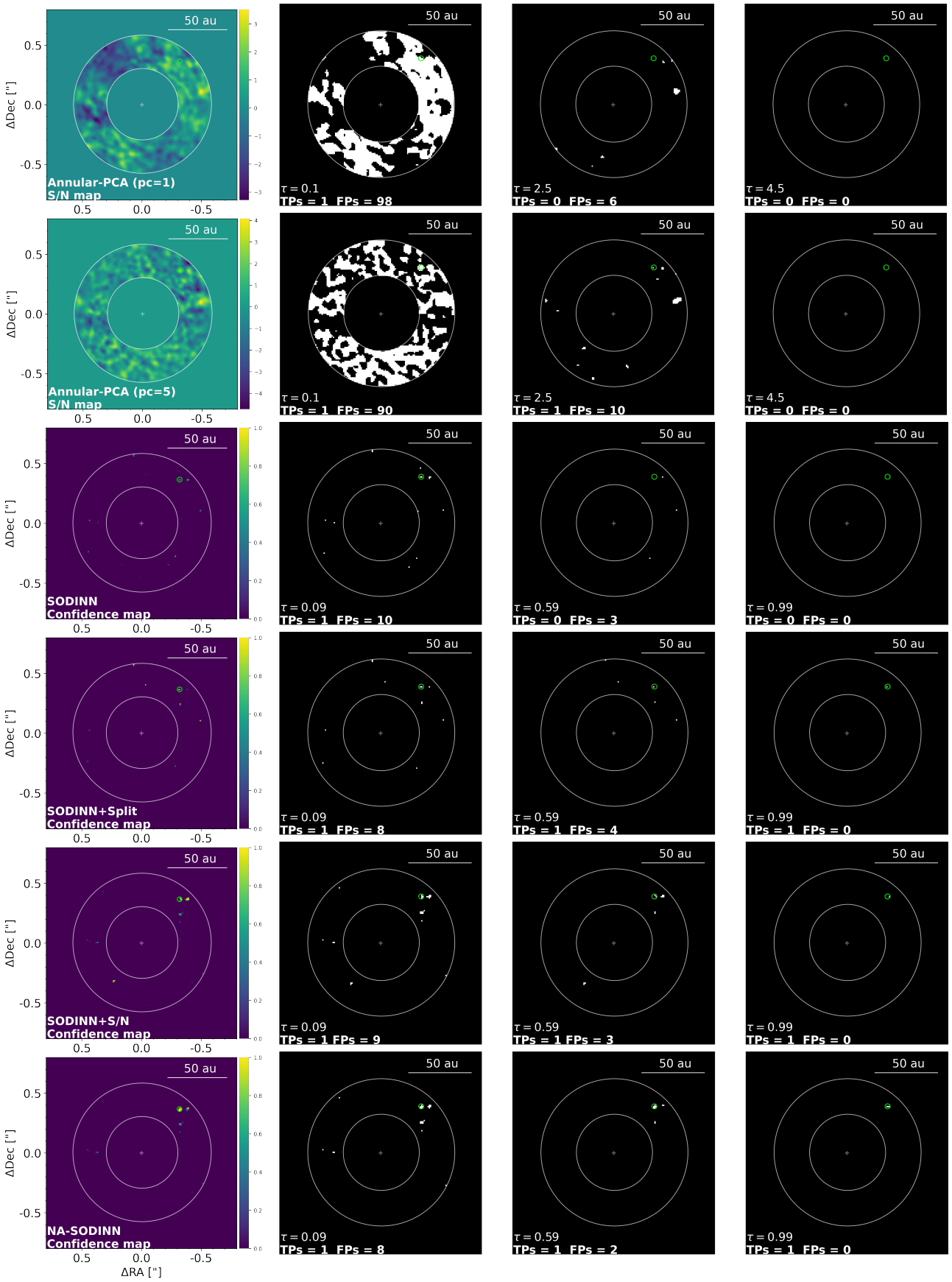


Fig. C.2: Same of Fig. C.1 for the regime **8-14** λ/D on **sph2**, where a fake companion has been injected with **S/N=0.89**.

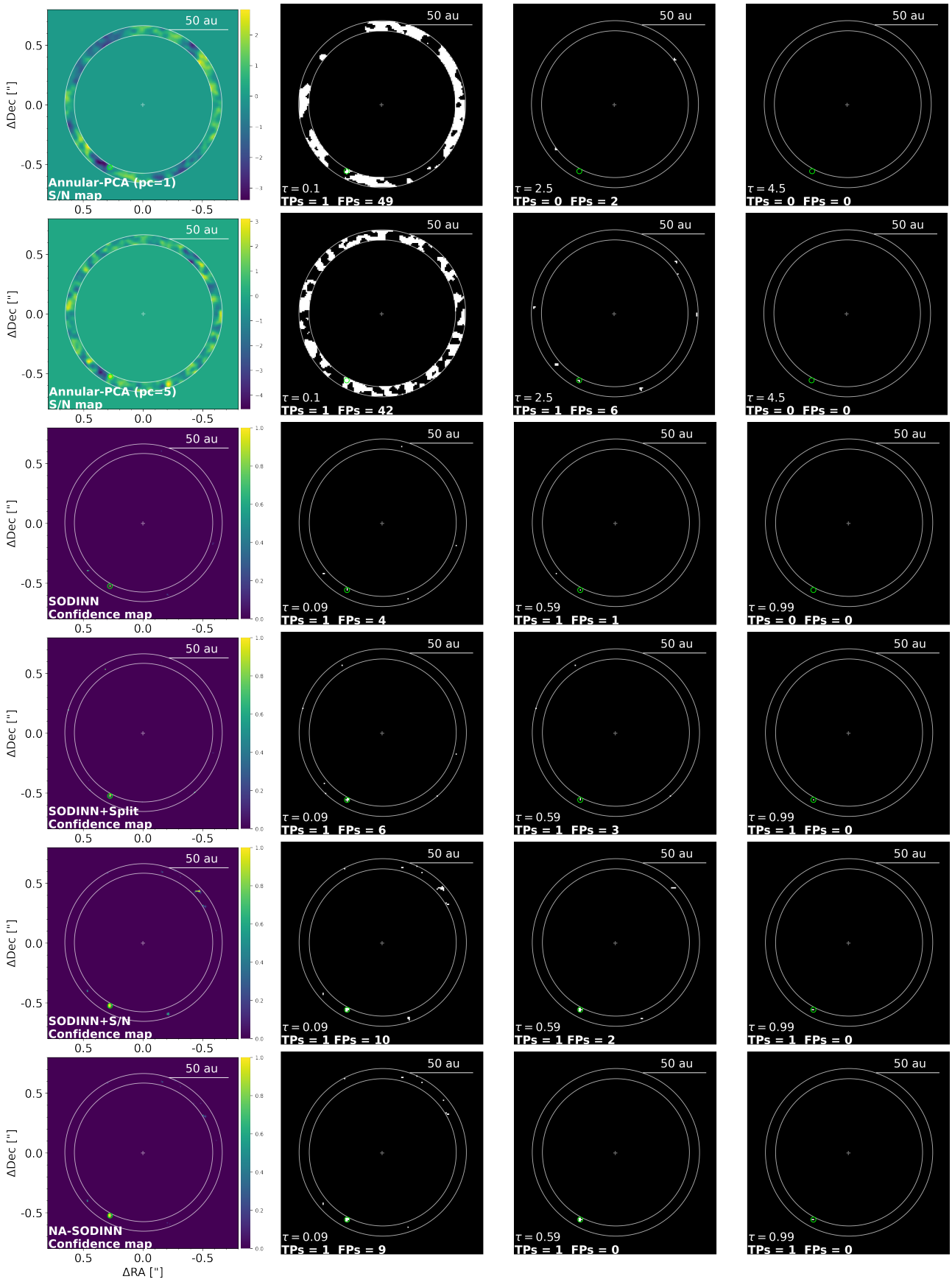


Fig. C.3: Same of Fig. C.1 for the regime **15-16** λ/D on **sph2**, where a fake companion has been injected with **S/N=0.78**.

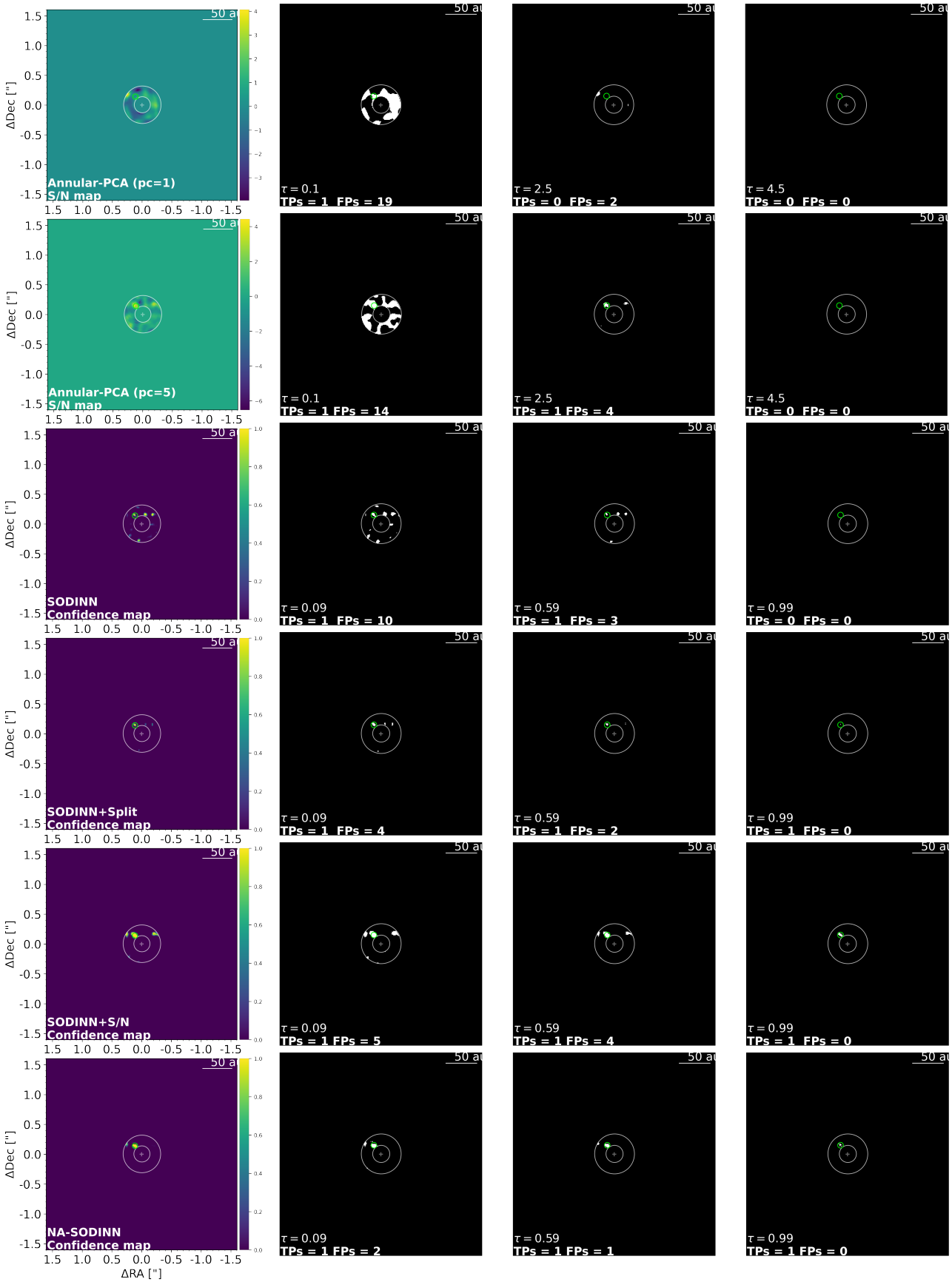


Fig. C.4: Same of Fig. C.1 for the regime **1-3** λ/D on **nrc3**, where a fake companion has been injected with **S/N=0.78**.

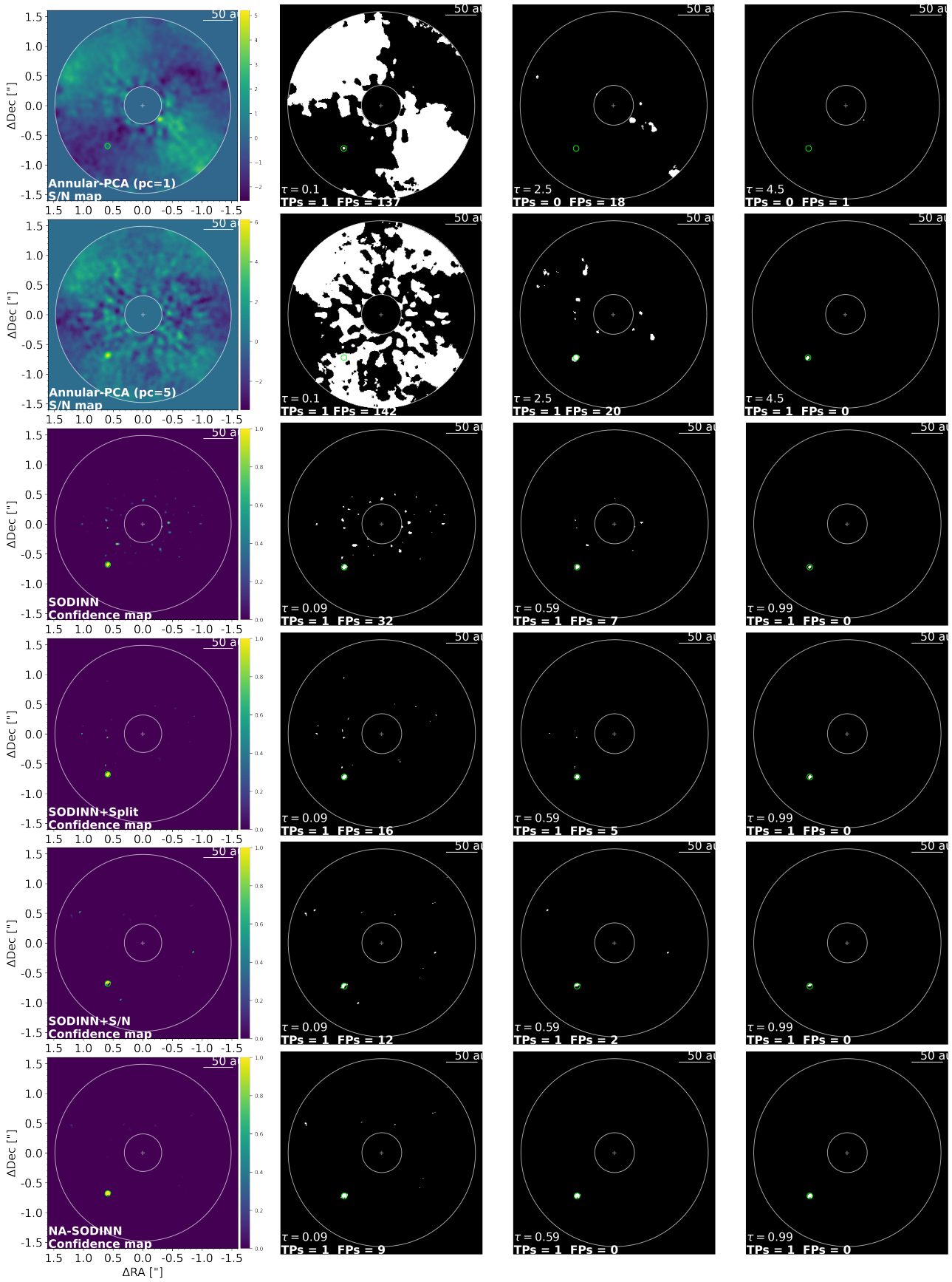


Fig. C.5: Same of Fig. C.1, but for the regime **4-16** λ/D on **nrc3**, where a fake companion has been injected with **S/N=0.84**.

Appendix D: Details of the EIDC metrics for NA-SODINN

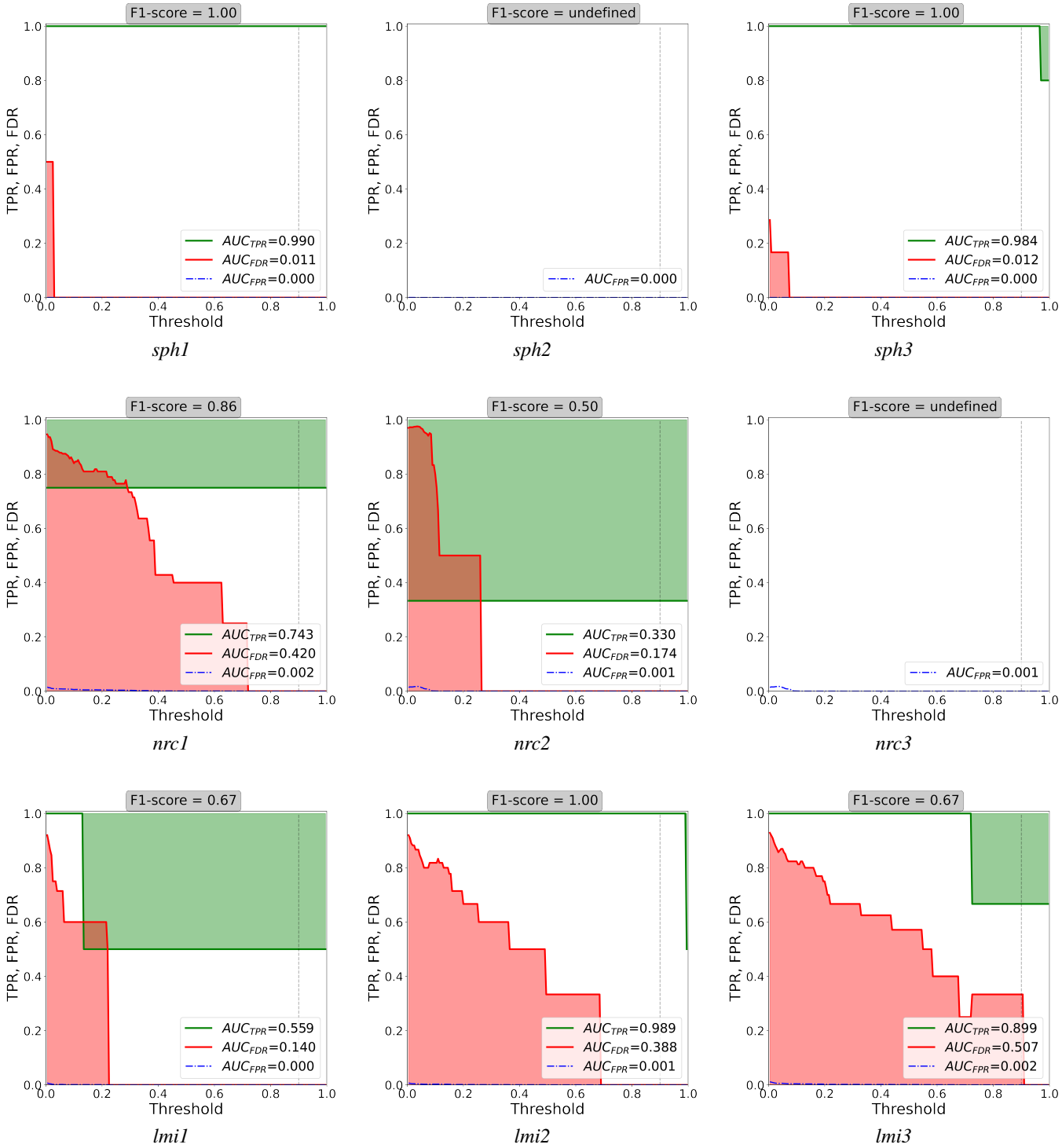


Fig. D.1: TPR, FDR, and FPR metrics computed from the confidence maps of Fig. 11 for a range of confidence thresholds varying from zero to one. Their respective AUCs are shown in each legend. The F1-score is computed at the submitted threshold on the challenge $\tau_{sub} = 0.9$ (vertical dashed line) and it is shown in the top of each subplot. When the dataset contains injections, TPR and FDR steply decrease with threshold, while FPR decreases monotonically. Thereby, an ideal algorithm would provide a TPR=1, FPR=0 and FDR=0 for any threshold and therefore, an $AUC_{TPR} = 1$, $AUC_{FPR} = 0$ and $AUC_{FDR} = 0$. However, when the dataset does not have injections, the FPR is the only metric that can be defined as it does not depend on TPs.