Análisis experimental del uso de *principal components analysis* (PCA) para la detección de exoplanetas

Lenin Valeria Rivas

16 de diciembre de 2025

1. Introducción

Entre las áreas más nuevas de la astronomía contemporánea, la detección de exoplanetas y la astrobiología se muestran en constante crecimiento, definiendo nuevas preguntas de investigación e innovando en nuevas metodologías de manera consistente. Entre las nuevas preguntas surge la posible existencia de vida fuera del Sistema Solar, buscando planetas similares a la Tierra en sistemas estelares externos. A pesar de que la detección de planetas símiles a la Tierra no ha sido posible con la tecnología actual, se han podido detectar planetas en sistemas estelares externos (exoplanetas) de mayor tamaño que la Tierra.

Entre los métodos más utilizados se encuentra el **tránsito planetario**, que caracteriza una disminución consistente y periódica de la curva de luminosidad de una estrella, producto del tránsito de un compañero entre ella y el observador, obstaculizando una porción de la luminosidad percibida. Sin embargo, este método está limitado a compañeros de estrellas que encuentren *face-in*, esto es, compañeros cuya órbita es paralela (o en su defecto es no perpendicular) a la línea de visión del observador. Esto implica que el compañero transitará eventualmente frente a la estrella, permitiendo caracterizar la disminución de luminosidad de la misma.

Desafortunadamente, este método no es efectivo para compañeros *face-on*, esto es, compañeros cuyo plano orbital es **perpendicular** a la línea de observación, implicando que no posee tránsito frente a la estrella y no es posible caracterizarlo mediante la curva de luminosidad. Esto limita la detección de estos compañeros a su propio brillo, el cual es mucho menor que el de la estrella ($\sim 10^{-9}$). Esto da origen al *high-contrast imaging* (HCI), que busca detectar exoplanetas dentro del desafío de imágenes de alto contraste

Dentro de las diversas técnicas utilizadas se encuentra el uno de **coronógrafos**, dispositivos físicos que se ubican en la cámara de detección para apantallar la luz de la estrella, que en otro caso opacaría completamente a su vecindad. Estos dispositivos cumplen su función de suprimir una porción importante de la luz de la estrella, revelando los elementos en su vecindad, pero a su vez genera un patrón de ruido denominado *speckles*, que se presenta de manera constante y cuasi-estática a lo largo de los cubos temporales.

Un desafío que está ganando tracción en tiempos recientes es buscar formas de identificar o modelar este patrón de ruido para eliminarlo de la imagen y facilitar la detección de potenciales exoplanetas. Entre los métodos que se han utilizado más ampliamente para este objetivo es la *principal components analysis* (PCA) que utiliza una reducción de dimensionalidad en sus *componentes principales*

más relevantes, buscando "filtrar" el patrón de ruido generado por la estrella y sustraerlo de la imagen.

Éste método no está exento de restricciones al momento de utilizarse, presentando problemas de *autosustracción* en casos de que el posible planeta a detectar coincida con puntos del patrón de ruido en momentos del cubos temporal, generando una sustracción adicional que puede, en el peor de los casos, eliminar completamente al potencial exoplaneta a detectar.

2. Preliminares

2.1. Angular Differential Imaging (ADI)

El *angular differential imaging* (ADI) es una rama del HCI que utiliza la rotación aparente del cielo mantentiendo la estrella fija en el cubo, para filtrar cualquier elemento móvil del ruido estático del instrumento.

Esto se logra fijando la cámara del instrumento en la estrella, dejando a la estrella misma estática al centro del cuadro y dejando que el fondo del cielo siga su desplazamiento. Esto generará un movimiento circular del medio alrededor de la estrella. En caso de presentarse algún planeta o estructura importante, se trasladará conforme se observan los distintos frames del cubo de datos.

Este movimiento angular está documentado y es entregado junto con el cubo *raw* de datos, como es el caso de VLT/SPHERE. Esto permite describir la posible trayectoria que el planeta recorrería si existiese, y también permite facilitar el proceso de "derotación" de las imágenes a lo largo del cubo.

2.2. Point spread-function (PSF)

Una *point-spread function* (PSF) describe el impacto que un instrumento de observación tiene al momento de detectar una fuente puntual, el cual se ve afectado por la infraestructura del instrumento, interferencias electrónicas del detector, fénomenos de difracción, entre otros. Esto se manifiesta en un efecto de "difuminación" o "deformación" en la imagen de ciencia, afectando a la correcta identificación de cuerpos dentro de la imagen.

La PSF comúnmente se asocia a la resolución del instrumento mismo: cuerpos de un tamaño menor al de la PSF que se encuentren muy juntos entre ellos serán imposibles de identificar entre sí, dado que se verán "fusionados" producto de la difuminación.

Dentro del contexto de HCI, una PSF del instrumento puede obtenerse capturando una imagen sin coronógrafo de la estrella del sistema a estudiar, cuidando que su intensidad no sature la cámara e inutilice la imagen final. Al ser el brillo de la estrella mucho mayor que su vecindad, en la imagen sólo se observará la estrella deformada por la PSF del instrumento.

2.3. Principal components analysis (PCA)

La *principal components analysis* (PCA) es una técnica estadística utilizada para **reducción de dimensionalidad**. Se utiliza principalmente para simplificar datasets complejos manteniendo la información más importante.

Los elementos de un dataset complejo están correlacionados unos de otros, los cuales son transfomados matemáticamente en un conjunto nuevo de variables ortogonales no correlacionadas llamadas

componentes principales (PCs). Estos componentes son ordenados por importancia en base a su varianza de manera decreciente. Esto significa que el PC1 es el componente con **mayor** varianza, el PC2 el componente con la segunda mayor varianza, y así sucesivamente.

Obteniendo solo los primeros componentes, se puede crear un modelo que represente las estructuras más dominantes de los datos, descartando aquellos que presenten menor varianza al ser considerados como ruido aleatorio.

En el contexto de HCI, la PCA fue aplicada en ADI para modelar el ruido y la dispersión del brillo generados por la estrella, en un método llamado *Karhunen-Loève Image Projection* (KLIP). Esta aplicación toma el cubo de imágenes del sistema como el dataset a reducir. Dado que los *speckles* son muy brillantes y cuasi-estáticos (varían ligeramente a lo largo del cubo), se interpretan como los componentes dominantes del dataset. Por lo tanto, los primeros PC que encuentre el algoritmo generarán de manera natural un modelo matemático de los speckles de la estrella.

Este modelo puede sustraerse de cada frame del cubo, resultando en solo los *outliers* del dataset. Dado que el planeta es mucho más tenue que la estrella y contribuye muy poca varianza al sistema, es ignorado por el algoritmo (idealmente). Sin embargo, si se opta por eliminar un número muy alto de PCs, el modelo comienza eliminar detalles que no forman de los *speckles*, incluido el posible planeta, generando **autosustracción**.

2.4. Vortex Image Processing (VIP)

La librería *Vortex Image Processing* (VIP) es una implementación de código abierto en Python, desarrollada especialmente para HCI de exoplanetas y discos circunstelares.

Entre las muchas funciones que entrega, facilita la inyección de planetas "falsos" con el objetivo de poder obtener un "*ground truth*" para realizar pruebas, a la vez que implementa diversos algoritmos de detección, incluyendo PCA en dos formatos: (i) *full-frame*, que utiliza los frames completos del cubo como referencias para extraer los PCs, y (ii) *annular*, que subdivide los frames del cubo en "anillos" aplicando la reducción de componentes a cada uno de manera iterativa, disminuyendo la posibilidad de autosustracción.

3. Experimentos

Todas las pruebas realizadas se encuentran en un repositorio de GitHub, ademas de los cubos utilizados para facilidad de reproducción.

Se realizará una comparación entre los métodos de PCA *full-frame* y PCA *annular* para observar sus diferencias en rendimiento, además de identificar en qué casos uno resulta mejor que el otro.

Para ello se utilizará un cubo de datos demostrativo, provisto por la documentación de VIP, que consiste de 61 frames de 101 x 101 píxeles del planeta Beta Pictoris, documentado en 2013. La ubicación del planeta es conocida, por lo tanto, se utilizará como referencia para las pruebas.

3.1. Detección del planeta real

En la documentación de VIP, se entrega la ubicación exacta del planeta Beta Pictorios con respecto a la estrella en el cubo (`x = 58.5 [pix], y = 35.5 [pix]`). Siendo la ubicación del planeta un valor conocido, se aplican ambos tipos de PCA en el cubo, variando el número de componentes principales utilizados para sustraer el ruido de *speckles*. Posteriormente, se extraen los valores de

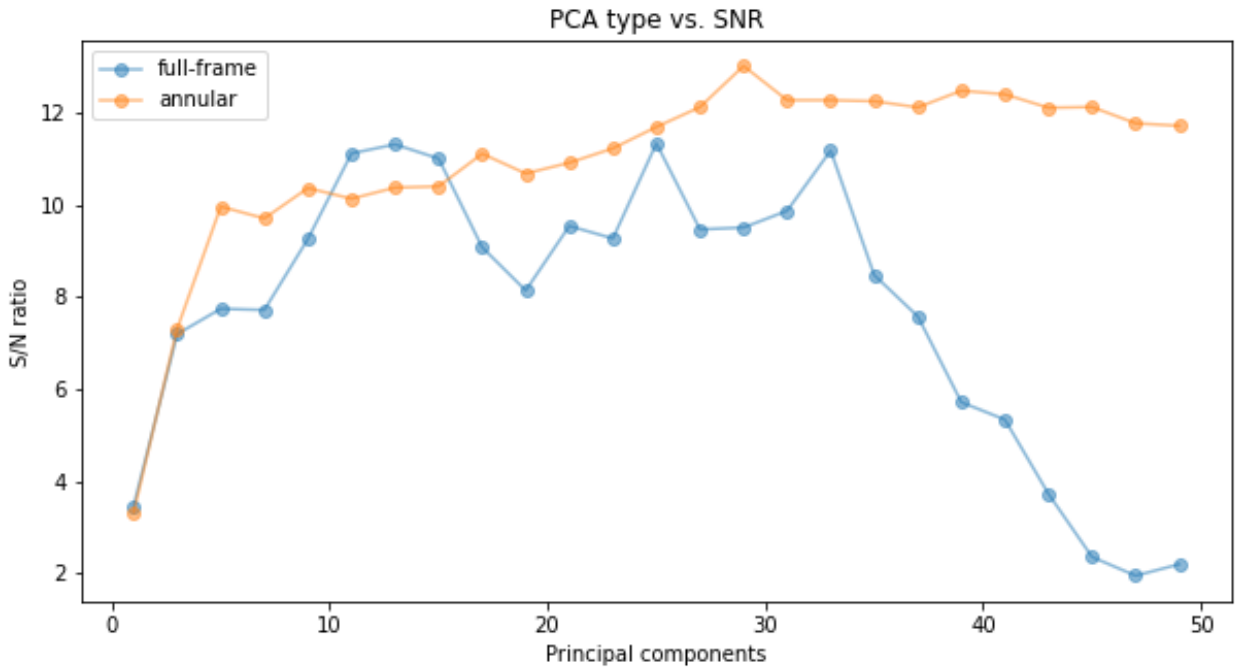


Figura 1: Valores de SNR vs. número de componentes principales (`ncomp`) utilizados. Notar la caída de SNR en PCA full-frame para un `ncomp` < 36. `full-frame` presenta un SNR mayor solo entre 10 a 14 componentes principales, luego se ve predominado por `annular`

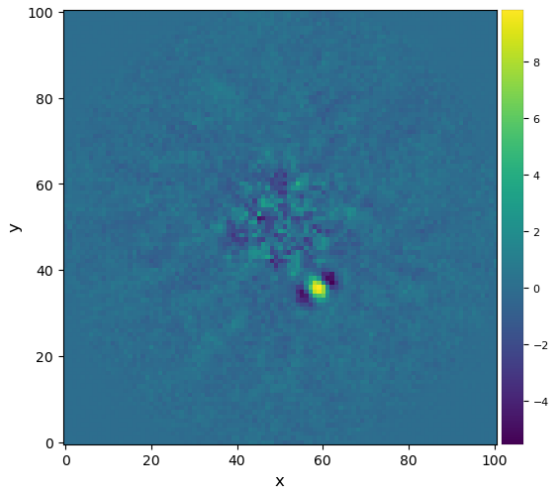
signal-to-noise ratio (SNR), una variable utilizada para representar todo valor de intensidad que difiere del fondo, considerado como ruido. Estos valores permiten identificar detecciones efectivas de posibles falsos positivos, a un mayor SNR, más confianza tiene la detección en la ubicación especificada.

En la imagen 1 se observan el rendimiento de ambos métodos para valores de componentes principales (`ncomp`) entre 2 y 48. Puede notarse que el valor de `ncomp` para el cual SNR es máximo es de 30 para `annular` y de 14 para `full-frame`, siendo el valor de `annular` mayor en general. También puede notarse como `full-frame` presenta una caída de SNR a partir de `ncomp = 36` mientras que `annular` se muestra consistente con valores de SNR ~ 12. Esto evidencia que `annular` se muestra más preciso que `full-frame` al momento de realizar la detección del planeta conociendo su ubicación. Sin embargo, `full-frame` requiere menos componentes principales para alcanzar su valor máximo, potencialmente disminuyendo el costo computacional del algoritmo.

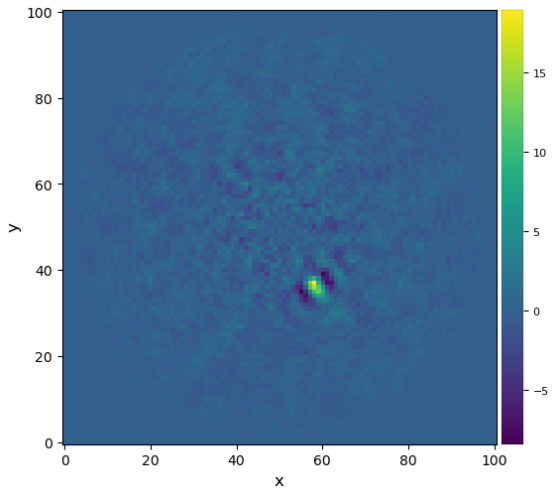
Este experimento por si mismo muestra una ventaja del metodo `annular` por sobre `full-frame`. Sin embargo, este es un caso aislado donde el planeta se encuentra cercano a la estrella, como puede observarse en la imagen 2. Esto da origen a la pregunta: ¿Es `annular` siempre mejor que `full-frame`?

3.2. Inyección de planetas

Posterior al expermiento anterior, se elimina el planeta del cubo mediante valores entregados por la documentación (mayores detalles en el GitHub). Luego, se inyectan 3 planetas en ubicaciones estratégicas y con brillos definidos, para evaluar como los algoritmos se comportan, conforme varía en número de componentes principales utilizados. Se asume que un planeta tenue, muy cercano a



(a) Imagen resultante después de aplicar PCA full-frame al cubo de datos con `ncomp` = 14.



(b) Imagen resultante después de aplicar PCA annular al cubo de datos con `ncomp` = 30.

Figura 2: Cubos resultantes luego de amplicar el número de componentes principales óptimo obtenido anteriormente. Notar como los valores de SNR para `annular` son considerablemente mayores a los de `full-frame`

la estrella, se verá autosustraido por ambos algoritmos, por lo tanto no se observará. Se toma como referencia el flujo de la estrella en el cubo como un valor arbitraio de 1000.

Los planetas inyectados son:

1. Un planeta **brillante** (`flevel = 400`) cercano a la estrella (`near-bright`), a una distancia radial de 15 píxeles del centro y una posición angular de 135°.
2. Un planeta **tenue** (`flevel = 200`) alejado de la estrella (`far-dim`), a una distancia radial de 45 pixeles del centro y una posición angular de 225°.
3. Un planeta **brillante** alejado de la estrella (`far-bright`), a una distancia radial del 45 píxeles del centro y una posición angular de 305°.

Cada planeta tiene una posición angular distinta que facilita la identificación visual de cada uno, el cual será un criterio para evaluar el nivel de autosustracción de los algoritmos.

Se aplicó cada uno de los métodos al cubo con estos 3 planetas inyectados, variando el número de componentes principales entre 10 y 60 (valor máximo del algoritmo para este cubo, reportado por el programa).

Para efectos de interpretación, también se extrajo el valor de SNR para cada una de las fuentes por cada valor de `ncomp`.

Dentro de los resultados obtenidos se aprecia como `full-frame` presenta una caída de SNR a un cierto número de `ncomp` para los tres casos, mientras que `annular` no presenta caídas de SNR significativas.

Las caídas de SNR, apreciables en la figura 4, para `full-frame` ocurren consistemente desde 10 hasta 40 componentes en `near-bright`, y aproximadamente a 50 componentes en `far-bright` y

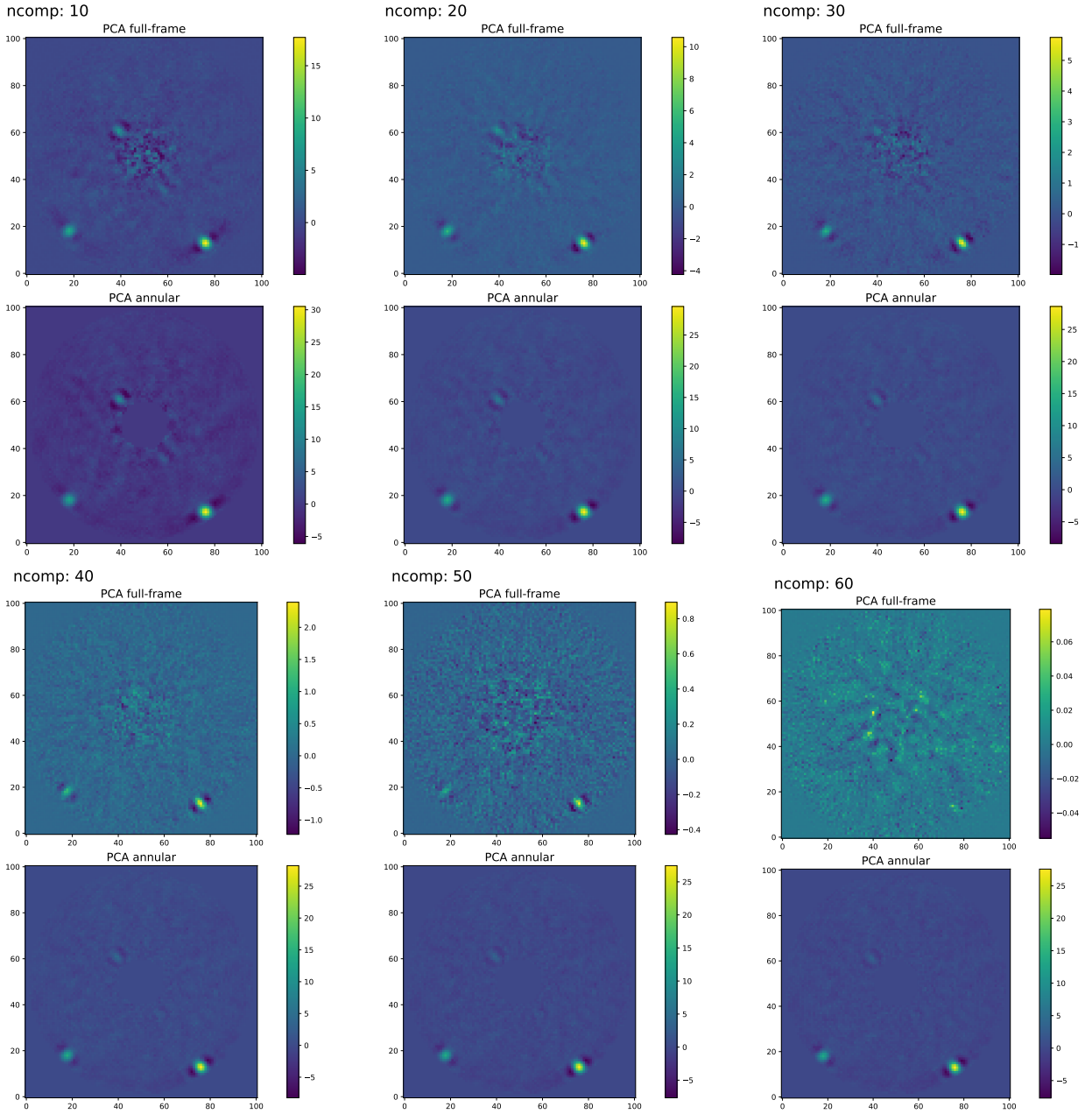


Figura 3: Imágenes resultantes luego de aplicar los algoritmos variando el número de componentes principales. Arriba y a la izquierda del centro se encuentra la fuente `near-bright`, abajo a la izquierda la fuente `far-dim` y abajo a la derecha la fuente `far-bright`. Notar como `near-bright` desaparece gradualmente conforme aumenta el número de componentes principales, harta ser imperceptible a `ncomp` = 30 en `full-frame`.

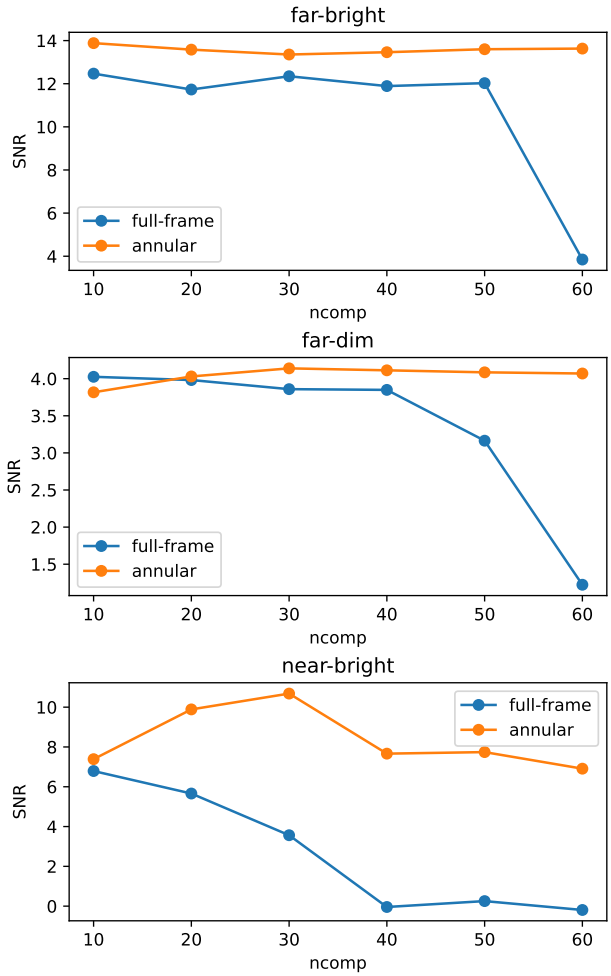


Figura 4: Valores de SNR para cada una de las fuentes. Notar como en `near-bright` el valor de SNR disminuye consistemente para `full-frame`, mientras que para `far-bright` y `far-dim` presenta caídas a ~ 50 componentes principales. Esto muestra una dependencia a la distancia del planeta en el sistema.

`far-dim`. Por otro lado, `annular` no muestra ninguna caída significativa en ningún caso, a excepción una pequeña oscilación entre 30 y 40 componentes en `near-bright`.

En cuanto a la apreciación visual de los planetas, apreciable en la figura 3, pueden identificarse claramente las fuentes alejadas al centro en todos los casos para `annular`, mientras que para `full-frame`, la fuente `far-dim` se empieza a perder a `ncomp` = 50, y para `ncomp` = 60 todas las fuentes se pierden.

Por otr lado, la fuente `near-bright` se identifica en `full-frame` hasta 30 componentes, y luego se pierde visualmente dentro del ruido. Para `annular`, la fuente no desaparece completamente, sin embargo se mantiene muy tenue conforme aumenta la cantidad de componentes utilizados.

3.3. Discusión

En base a los resultados observados, se deducen las siguientes conclusiones:

- `full-frame` depende de la distancia a la que se encuentra el planeta, independiente de su intensidad. Si un planeta se encuentra muy alejado de la estrella, independiente de si es muy brillante, se perderá a un número muy elevado de componentes principales.
- `full-frame` es muy sensible a la cantidad de componentes utilizados, contrario a `annular`. Sobre 60 componentes se pierde toda la información de este cubo en particular, mientras que `annular` pareciera conservar las estructuras, independiente de la cantidad de componentes que utilice para sustraer el ruido.
- `annular` no depende de la distancia ni de la intensidad de brillo del planeta, apreciable en las imágenes 1 y 4, manteniendo un valor de SNR (el cual si depende del brillo del planeta y su contraste con la estrella) consistente conforme aumenta el número de componentes.

Por lo tanto se puede aseverar con más seguridad y en base a los experimentos realizados que `annular` es **mejor** que `full-frame` en términos de precisión de detección y eliminación de ruido por *speckles*.

3.4. Conclusiones

Dentro de todo lo obtenido, se realizó un análisis exploratorio básico de los formatos de PCA entregados por la librería VIP, en el marco de la detección de exoplanetas. Adicionalmente, se deja abierta la oportunidad de ahonda mucho más en la implementación de estos algoritmos, dado que se utilizaron las funciones más básicas disponibles, sin alterar muchos argumentos opcionales que la documentación especifica.

Entre los temas que no se observaron en este proyecto está el costo computacional de los algoritmos. La documentación no especifica en gran detalle lo que realizan de fondo, un análisis del código fuente sería necesario para obtener resultados sólidos. En general, al ser `annular` un algoritmo iterativo de tipo *divide and conquer*, resulta más ligero a nivel de memoria, pero presenta un mayor costo computacional, contrario a `full-frame`, que ingresa los frames completos a la memoria, pero resulta más rapido en cuanto a poder computacional.

El proyecto, a grandes rasgos, cubrió los objetivos propuestos en un principio, evaluando de manera experimental el uso de PCA en el área de HCI. Sin embargo, la utilización de ambos formatos de PCA disponibles y su contrastación en distintos casos es algo nuevo, que ayudó a estructurar de manera más sólida el proyecto y la experimentación.

Finalmente, este proyecto resultó de gran ayuda como introducción a los algoritmos de HCI, fijando una base para un análisis más complejo en el futuro, con objetivos de poder desarrollar algoritmos dentro del área. Esto mediante un acercamiento más directo al uso de las implementaciones, conocidas en teoría pero que nunca habia visto en funcionamiento.