Jennifer Studer
studerje@student.ethz.ch

To be determined

Master Thesis

Exoplanets and Habitability
Institute of Particle Physics and Astrophysics, D-PHYS
Swiss Federal Institute of Technology (ETH) Zurich

Supervision

Hans Martin Schmid
Christian Tschudi

March 6, 2022

Abstract

Contents

1 Introduction

2 Aperture Photometry

In order to determine the flux of different objects in astrophysics, like stars and planets, aperture photometry can be used. This method sums the counts of the pixels inside a certain aperture around the star. In our case this aperture is usually a circle. In order to account for the background noise an annulus around the aperture is taken and the mean of the summed up pixels inside the annulus is subtracted from the apertures pixels. The flux of this aperture is then given by [3]

$$F_{ap} = F_{tot} - n_{px}\langle F_{bg}\rangle, \tag{1}$$

where F_{tot} is the total flux inside the aperture (sum up the pixel values inside the aperture), n_{px} is the number of pixels inside the aperture and $\langle F_{bg}\rangle$ is the mean background per pixel. This mean background per pixel is defined through the annulus and calculated from

$$\langle F_{bg}\rangle = \frac{1}{m}\sum_{i=1}^{m} c_i, \tag{2}$$

where m is the number of pixels in the annulus and c_i the respective pixel value. Figure 1 shows an example for a possible aperture and an annulus around a star, which can be used to do an aperture photometry.

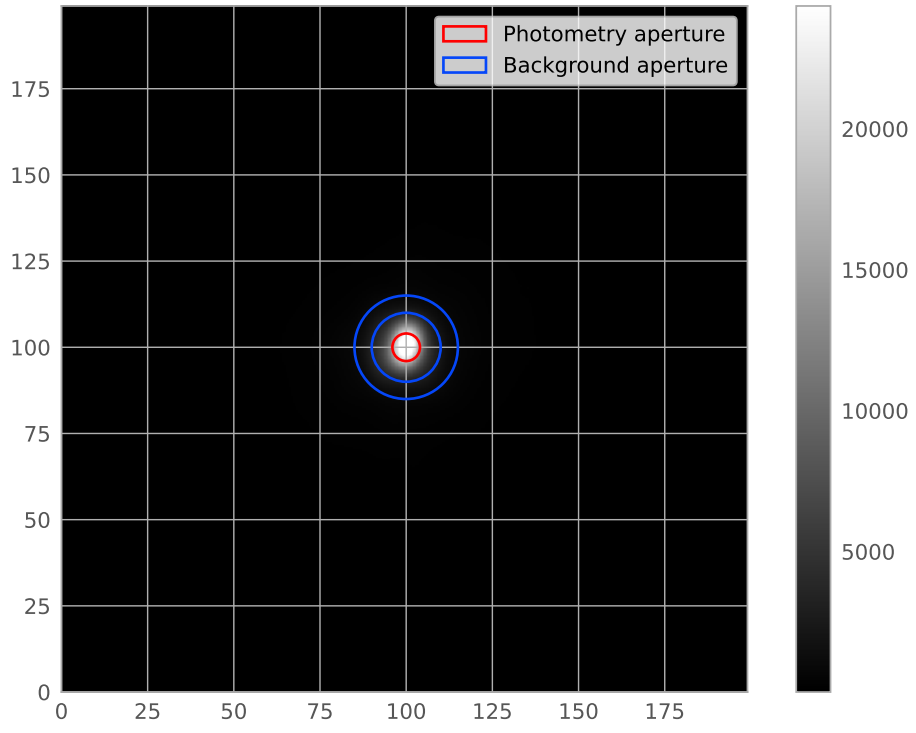


Figure 1: An aperture photometry for the star in the center, where the red circle indicates the aperture used and the two blue circles define the annulus used for the background subtraction.

From the figure you can see that we chose the annulus not directly after the aperture, but this is just one way to do it. One could also choose the annulus directly after the aperture or choose a different distance between the annulus and the aperture. However the annulus should give a good

approximation for the background inside the aperture and therefore it should not be too far away from the aperture. We chose a small distance between the aperture and the annulus of 4 pixels, because we wanted that as little starlight (in this case) as possible is included in the annulus. If we plot the counts per pixels which are included in a certain radius around the star, as it is done in figure 2 we see that after around 10 pixels the increase is decreasing rapidly. That is where there is only little starlight left. This is also why we choose the annulus to go from a radius of 10 to 15 pixels.

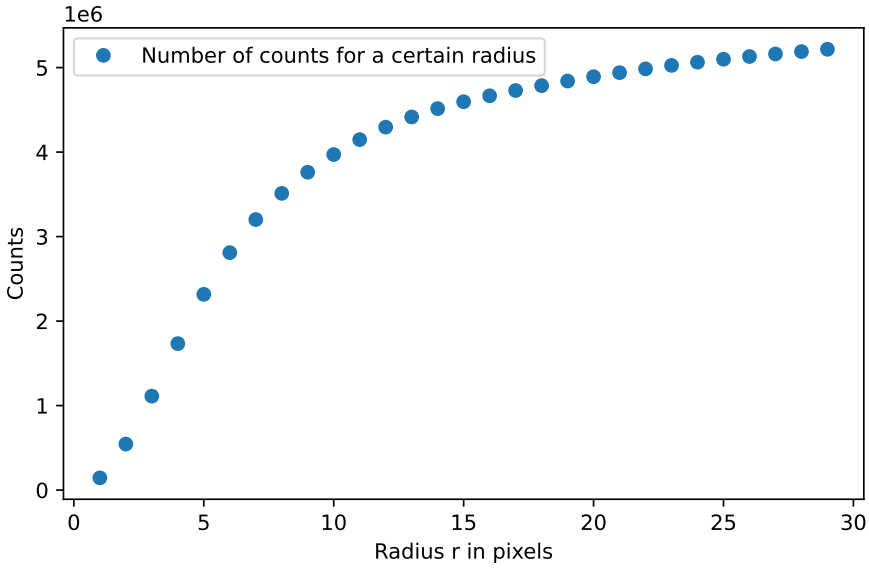


Figure 2: The total flux of the star is calculated for different radii and plotted. This shows that after a radius of 10 pixels the contribution from the star is almost gone.

Additionally we choose the radius of the aperture to be 6 pixels.

2.1 Aperture photometry and ghosts

In order to detect exoplanets one can use aperture photometry. In this subsection we are going to do these steps for the two ghosts which we have in our data from the circumstellar disk HD142527, in order to demonstrate how this could be done for an exoplanet and to learn more about ghosts. A ghost is a copy of the star, which is created by the back-reflection of the star on optical components of the telescope. Figure 3 is an image of HD142527, where the two ghosts are indicated. We see that the ghost on the top right (we will call this ghost 1) is brighter than the ghost on the bottom left (ghost 2).

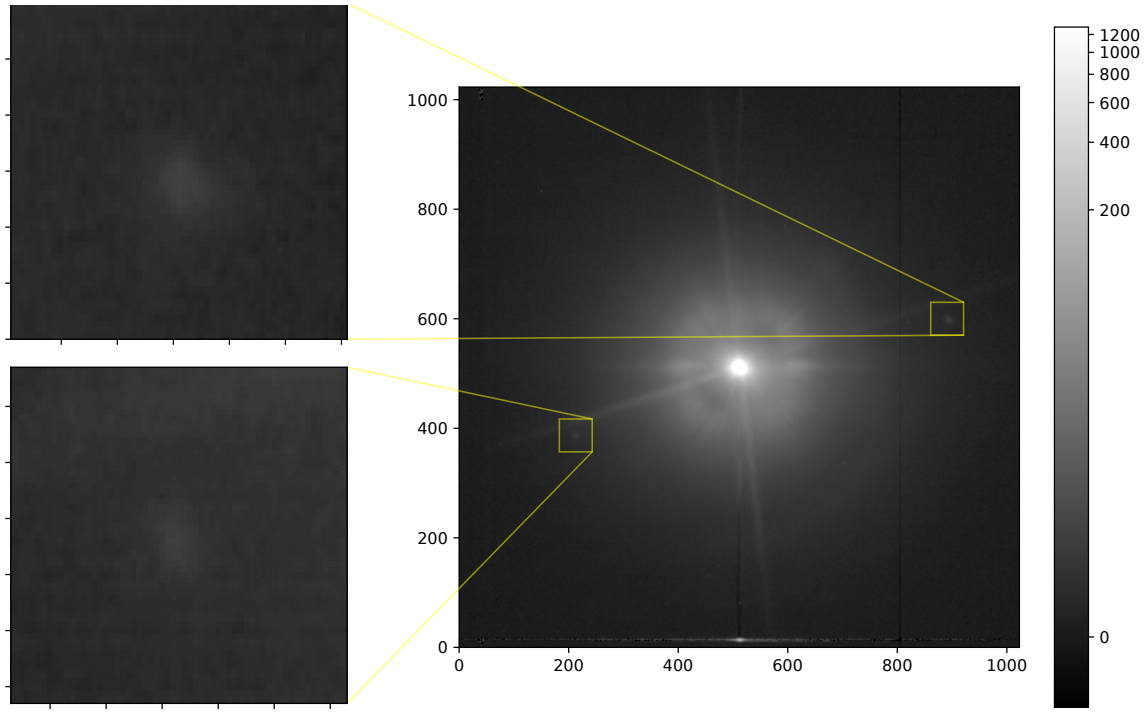


Figure 3: An image of the circumstellar disk HD142527, where the two ghosts are indicated.

If we want to confirm the signal from our ghost or also from other objects like exoplanets, which usually can not be seen by eye, we use the signal to noise ratio S/N. This means we do aperture photometry for several points around the star which are at the same separation from our star as the ghost (or exoplanet etc.), as in figure 4. From this we can calculate the standard deviation σ from the mean of all aperture fluxes and the signal to noise ratio. If the signal to noise ration is larger than 3σ, this means that we have a source (ghost, exoplanet) at this position.

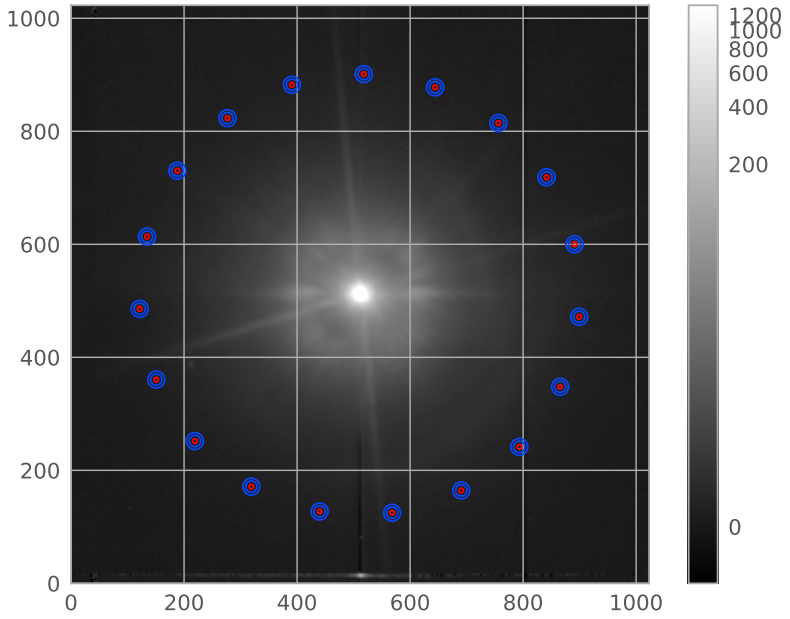


Figure 4: In order to confirm the signal from the ghost, we do aperture photometry for several points around the star which are at the same distance from the star as the ghost. We then calculate the standard deviation of all the aperture fluxes and from there find the signal to noise of the ghost's aperture. If the signal to noise is larger than the standard deviation, the position of the ghost is confirmed.

The standard deviation is calculated by

$$\sigma = \sqrt{\frac{1}{k-2}\sum_{ap=2}^{k}(F_{ap} - F_{mean})^2}, \tag{3}$$

where the aperture of the ghost is at $ap = 1$, k is the number of apertures and F_{mean} is the mean flux of the apertures (without the aperture of the ghost) given by

$$F_{mean} = \frac{\sum_{ap=2}^{k} F_{ap}}{k-1}. \tag{4}$$

From this we can find the signal to noise ratio as

$$S/N = \frac{F_1 - F_{mean}}{\sigma}. \tag{5}$$

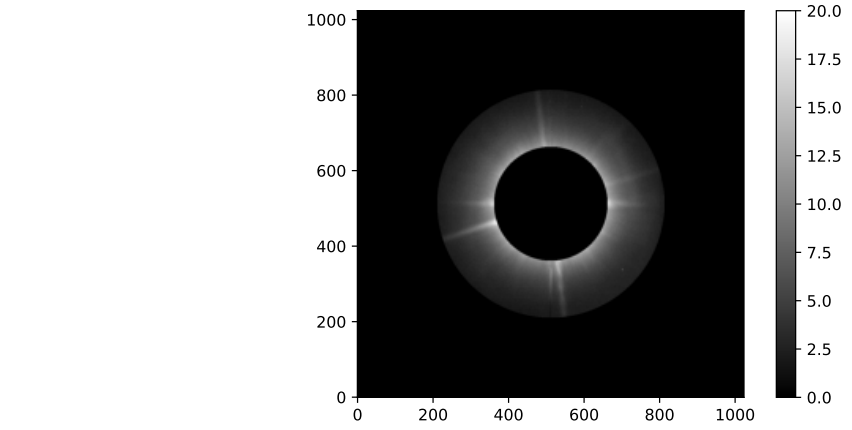
The results we get from our data about the ghosts are shown in table 1. It confirms that ghost one is where we expected it to be, since its the signal to noise is larger than the 3σ. For ghost 2 this is not the case. The signal to noise is larger than the standard deviation, which tells us that there might be an object, but the confidence interval is only 2σ. This is not enough to confirm the detection.
We also want to compare the intensity of the ghosts to the intensity of the star. For this we calculate the ratio between the aperture flux of the ghost and the star. We find that the ghosts are about 10^{-4} times less bright then the star.

	Ghost 1	**Ghost 2**
S/N	33.9	16.5
σ	7.9	9.8
Ratio	$1.1 \pm 0.4 \cdot 10^{-4}$	$7 \pm 1 \cdot 10^{-5}$

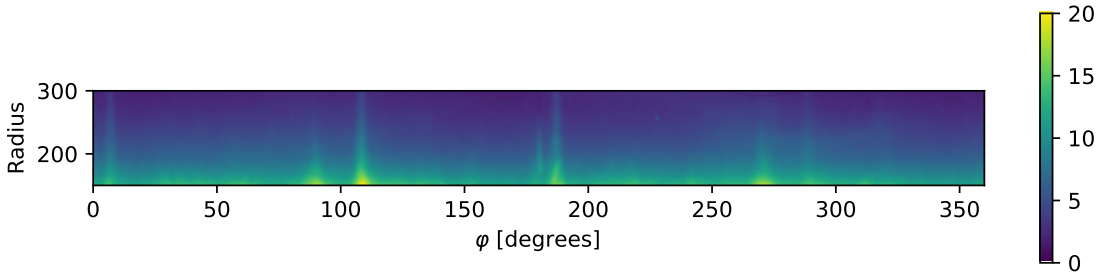
Table 1: Signal to Noise of the ghosts and their brightness with respect to the star. The standard deviation σ is given in counts.

3 Transformation to the r-φ plane

We have now seen that the ghosts are about 10^{-4} times less bright then their star, but potential exoplanets are even less bright. A Jupiter like exoplanet would be 10^{-9} less bright (reflecting light) and an Earth like even only $2 \cdot 10^{-10}$. Therefore the data must be really sensitive, with a high contrast and a high spatial resolution. As we can see in figure 3 the star produces strong spiders and speckles which have similar brightness as the ghosts or are even brighter. Therefore exoplanets which are situated in this regimes cannot be detected, unless we are able to take out the signals of the spiders and speckles without taking away other signals, like the ones from exoplanets.
If we take a closer look at figure 3 we observe that the structure of the spiders and the speckles are radially oriented around the star. In order to get rid of this effects it might be a good idea to transform the image into the r-phi plane, where we define the star to be at radius zero. After the transformation the spiders and speckles are distributed along the phi axis and they become weaker along the r axis. The image before and after the warping is shown in figure 5, where we only warped the part of the image which is within radius 150 to 300 pixels.



(a)



(b)

Figure 5: We mask the region (radius=150-300 pixels) which will be warped to the r-φ plane (a). The resulting warped image (b), where we used spline interpolation.

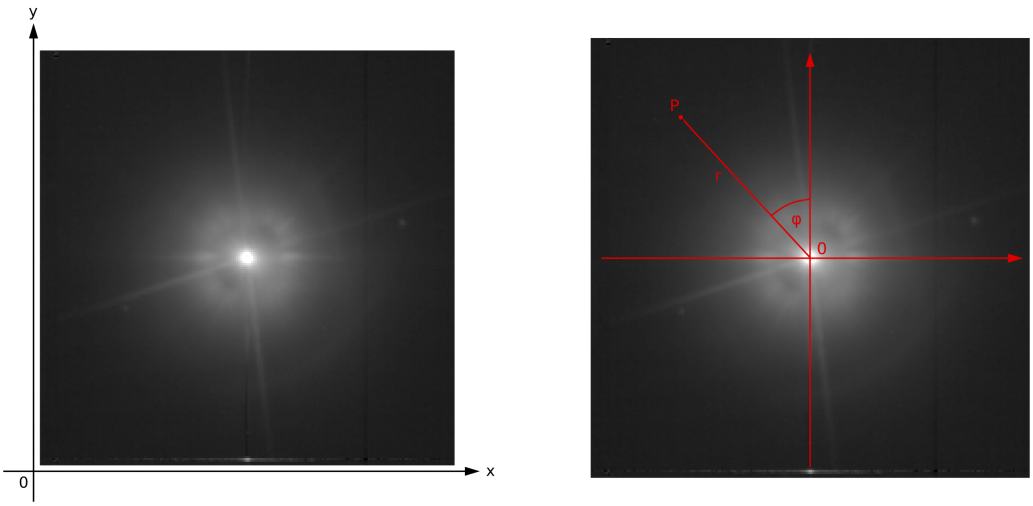
Lets have a look at how one can transform the image into the r-phi plane. This kind of transformation is called image warping in image processing. In our case the image warping is based on a specific transformation, namely the transformation from Cartesian to polar coordinates and is therefore also called polar-cartesian distortion.
In general an image warping is based on a transformation $T : \mathbb{R}^2 \to \mathbb{R}^2$, such that

$$\vec{x} \mapsto \vec{u} = \begin{pmatrix} T_u(x, y) \\ T_v(x, y) \end{pmatrix}, \tag{6}$$

where $\vec{x} = \begin{pmatrix} x \\ y \end{pmatrix}$ and $\vec{u} = \begin{pmatrix} u \\ v \end{pmatrix}$. When we want to warp an image f into an image g we do the following calculation

$$g(\vec{u}) = g(T(\vec{x})) = f(\vec{x}). \tag{7}$$

This means that at pixel $\vec{u}$ the computed image g has the same intensity as the original image f at pixel $\vec{x}$. [2]
When we want to describe the position of a pixel we use Cartesian coordinates, the way it is shown in figure 6 (a), but there are also other ways to describe the location of the pixels. Figure 6 (b) shows the image with a polar coordinate system, where we chose the origin to be in the center (where the stars position is).



(a) (b)

Figure 6: To describe the location of pixels in an image one usually uses Cartesian coordinates (a), but we can also use polar coordinates (b).

In order to change from Cartesian coordinates $x, y \in \mathbb{R}$ to polar coordinates $r \in [0, \infty)$, $\varphi \in [0, 2\pi)$ one needs to do the following calculations:

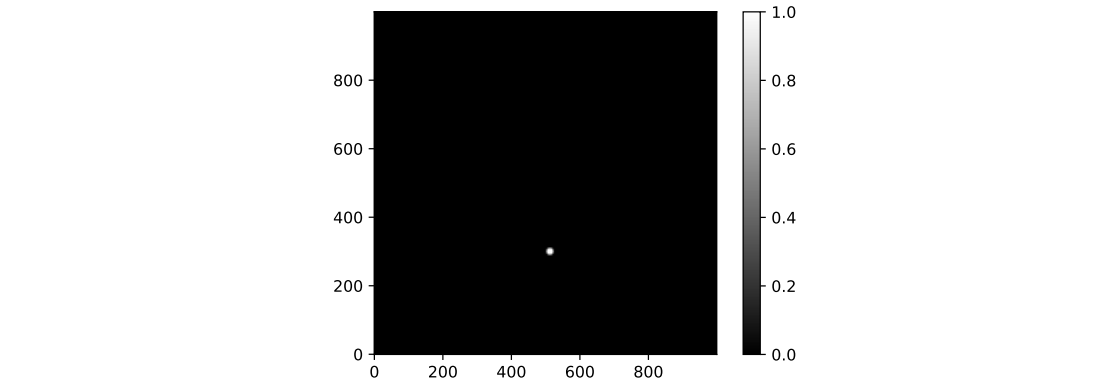
$$r = \sqrt{x^2 + y^2} \qquad \varphi = \arctan\left(\frac{-x}{y}\right). \tag{8}$$

For the back transformation from polar to Cartesian coordinates we compute

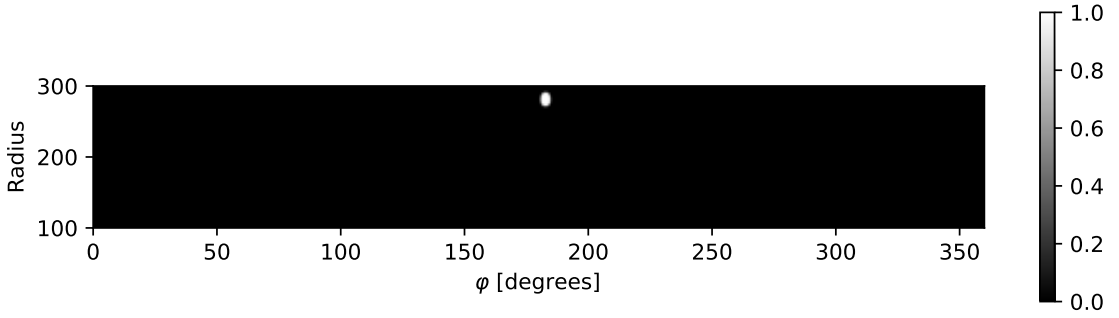
$$x = -r\sin(\varphi) \qquad y = r\cos(\varphi). \tag{9}$$

The transformation of the pixels from Cartesian to polar coordinates results in pixels which are not squares. However for our resulting image in the r-φ plane, we need square pixels. This pixel distortion happens because the number of pixels included at a certain radius away from the center increases proportionally to the radius. This means that there are twice as many pixels at a radius of 100 (pixels) than at radius 50. We now have two options after we have defined the grid of our r-φ plane. Either we choose the grid such that at $r = r_{max}$ every pixel in the grid corresponds to a pixel in the original image and for smaller radii we have pixel which are empty. Or we choose the size of the grid differently and use interpolation to assign an appropriate value to each pixel in the r-φ plane. We decided to use interpolation, since we cannot work with the data if there are empty pixels. In figure 5 we have already seen an example for this transformation using spline interpolation.
Now lets think about which length the grid should have. Since our main interest is to find round objects like exoplanets, it would be beneficial if round objects are still round after the transformation. In order for this to be satisfied, the number of pixels at the radius position of the object has to stay unchanged. We therefore decided to choose the radius range (the φ angle goes from 0 to 360 degrees and thus covers the whole circle), such that the object we want to examine is in the middle and the grid size corresponds to the length of the radius range, i.e. for $r_{min} = 100$ and $r_{max} = 300$ the length of the grid the radius direction is $r_{len} = r_{min} - r_{max} = 200$. The resulting grid size in φ direction depends then on the chosen radius range. Namely such that: $\varphi_{len} = 2\pi \cdot (r_{min} + \frac{r_{len}}{2})$. Figure 7 illustrates the effect of this transformation onto circles in the original image (a). In the center of the warped image at $r = 200$ the number of pixels is unchanged by the transformation, this means no interpolation nor averaging was needed. Therefore circles stay circles, if they are placed in the middle of the warped image, see figure 7 (c). If we go to larger radii the number of pixels per radius (in original image) increases. Since in the warped image all radii have the same

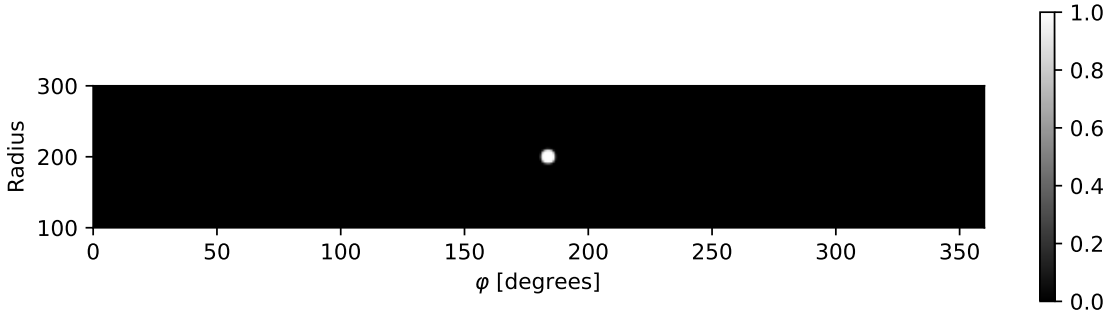
number of pixels r_{len} the transformation averages the information in the original pixels into fewer pixels and the circle becomes elliptic like in its shown in figure 7 (b). The opposite effect happens if we go to smaller radii where the number of pixels per radii (in original image) decreases and interpolation is needed to distribute the original information to an enlarged number of pixels. This results also in an elliptic shape, but with the semi-major axis now in the φ direction, see figure 7 (d).



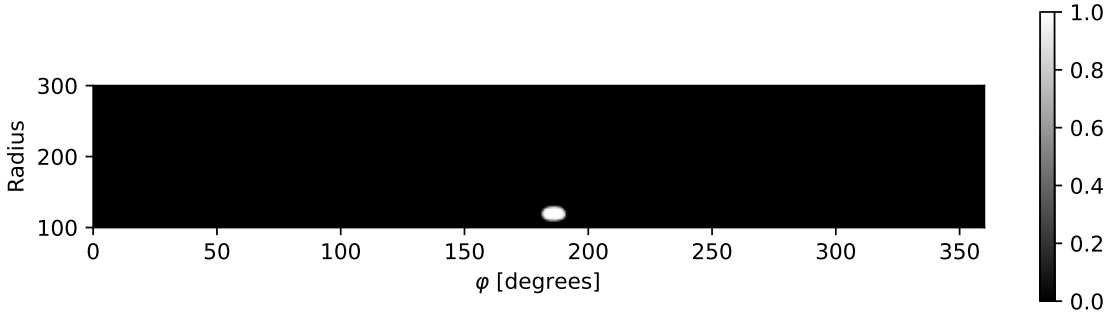
(a)



(b)



(c)



(d)

Figure 7: The warping is defined such that a point in the original image (a) is left unchanged, if it is in the middle of the radius range of the transformed image (c). Otherwise the point will become an ellipse (b) and (d).

In order to perform the transformation to the r-φ plane, we need to define the new shape of the warped image, as discussed before. From this we then define the new pixel grid in polar coordinates and then assign the corresponding Cartesian coordinate values. With this information it is already possible to map the values of the pixels in the original image to the warped one. In our program (shown below [2]) we used the python package scipy.ndimage.map_coordinates() which uses cubic spline interpolation to do the mapping.
As we already mentioned interpolation is used to assign a value to every pixel in the new frame by using the information from the old frame. Or in other words the wholes (where we have no information) are closed by cleverly inventing new values for these wholes through the observation of the information in the neighborhood of the whole. The used spline interpolation fits polynomials to the known values (in our case third order polynomials) in the neighborhood and takes then the values given by the polynomials. We chose this method, because we received the best results with it, but one can also use different interpolation methods like nearest neighbor interpolation or the bilinear interpolation.

```
def to_rphi_plane(image, im_shape, r_min, r_max):
    """
    Warping to r-phi plane.

    Parameters
    ----------
    image : float32, np.array
        An intensity image.
    im_shape : (int, int)
        Shape of image f.
    r_min : int
        Inner radius.
    r_max : int
        Outer radius.

    Returns
    -------
    warped : float32, np.array
        r-phi plane image.

    """
    # Define the shape of the resulting warped image
    r_len = r_max - r_min
    phi_len = int(2*np.pi*(r_min + r_len/2))

    # Define the new grid
    rs, phis = np.meshgrid(np.linspace(r_min, r_max, r_len),
                           np.linspace(0, 2*np.pi, phi_len), sparse=True)

    # Assign the corresponding Cartesian coordinates to the new grid
    xs, ys = rphi_to_xy(rs, phis)
    xs, ys = xs + im_shape[0]/2 -  1, ys + im_shape[1]/2 - 1
    xs, ys = xs.reshape(-1), ys.reshape(-1)
    coords = np.vstack((ys, xs))

    # Create the warped image with spline interpolation 3th order
    warped = scipy.ndimage.map_coordinates(image, coords, order=3)
    warped = g.reshape(phi_len, r_len)

    return warped
```

It is also possible to transform the image back to the Cartesian coordinate map. By simply doing a warping of the image in the r-φ plane back to the original x-y plane.
Lets go back to our example with the circle and have a look if the aperture flux of the circle is preserved trough the warping of the image. We find that the circle has an aperture flux of 306, after the transformation to the r-φ plane this aperture flux is 302. So the aperture flux does not change much through the transformation. After the back transformation we get an aperture flux of 306

which is the same as in the original image. Still we also need to check for this in our data, where we have smooth transitions. We do so by using ghost 1 which is at a radius of 390 pixels in the data of HD142527. For the flux of the aperture in the image we get 231.5. After a transformation to the r-φ plane where we chose $r_{min} = 290$ and $r_{max} = 490$ the flux of the aperture was 232.1. This is a change of 0.5%. So we can say that the transformation conserves the aperture flux which is a really good and important property.

4 Radial Intensity drop off

Due to the star in the center of the image, there is an intensity decrease in radial direction. This light coming from the star makes it harder to find other objects and structures close to it. Therefore we want to get rid it. This is a lot easier in the r-φ plane, where the intensity decreases is parallel to the r axis as we can see in figure 5 b) and this dropp of can be described by an exponential decrease. In figure 8 the mean angular intensity values are plotted against the radius and the data is fit by an exponential.

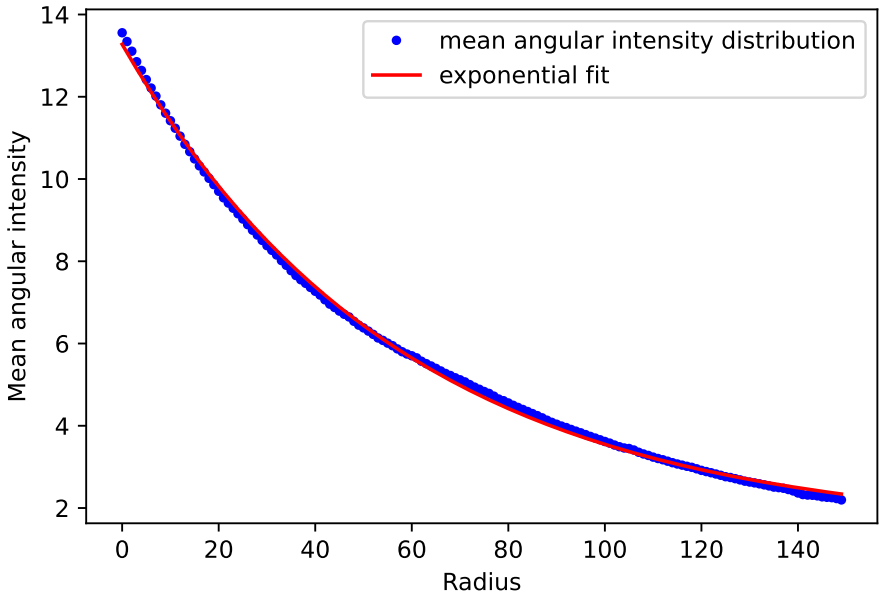
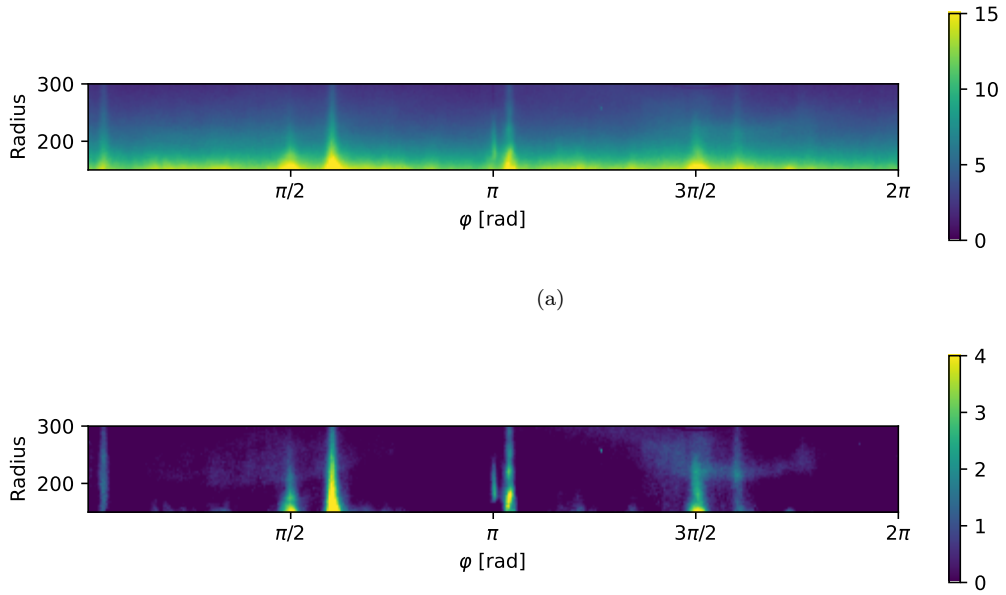


Figure 8: The intensity drop off due to the light from the star at radius zero can be described by an exponential fit.

The exponential fit describes the intensity drop off in a good way and by subtracting it from the r-φ plane image we get an image where the mean intensity in radial direction is more or less constant. Figure 9 shows the effect of this subtraction and we call the resulting image the flatten image.



(a)

(b)

Figure 9: After the image transformation into the r-φ plane (a), the image is flattened by subtracting the intensity which comes from the star at radius zero (b).

We insert a model planet (a faint copy of the star) into the image before the flattening of the image to make sure, that the aperture flux is not affected by the flattening. We find that the aperture flux change due to the flattening is only 0.03%. So we can say that the aperture flux is not affected by this procedure.

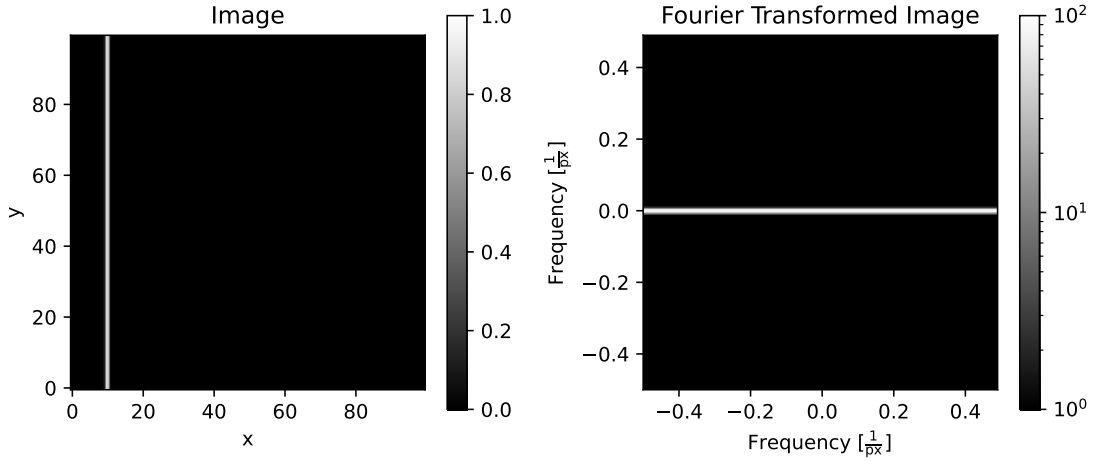
5 Fourier Transformation

Our goal is to decrease the intensity of the spiders and speckles in the data. In our data from HD142527 the main disturbing effects are the spiders. With the transformation to the r-φ plane the spiders now are parallel to the r-axis. When comparing different data sets from the same observation we find that the spiders change there position, but the distance between them stays the same. It is a periodic pattern. This brings up the idea, if the spiders are represented by a set of frequencies in the frequency plane. If this is the case, the suppression of some of the frequencies in the frequency plane would result in a suppression of the spiders in the image plane. The advantage of this method would be that it could be applied to the complete data set and one can ignore the fact that the spiders wander along the φ-axis. Also we should be able to suppress the spiders without destroying the information below them.
In the following we are going to investigate the properties of a Fourier transformation on some specific image structure as lines, beams and point-like sources.

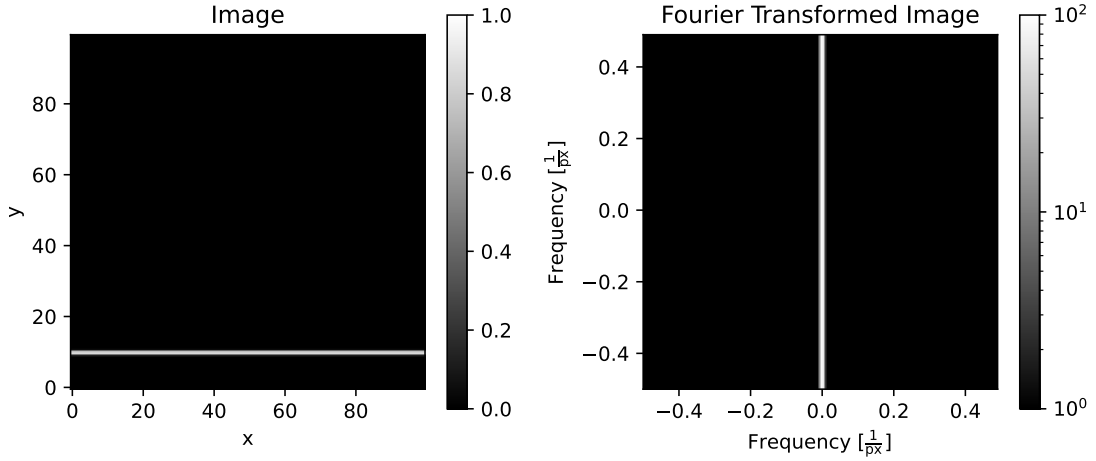
5.1 Lines and Beams

We first want to investigate the effect of some simple signals in the image plane on the frequency plane. We choose these signals similar to the shape of the spiders or to the shape of a potential exoplanet, with the goal to identify similar characteristics in the frequency plane of the data.
Firstly, we transform a single line (has the width of one pixel) in vertical or horizontal direction to the Fourier plane. A single line in vertical direction (y-axis) means that we have a constant signal

along the y-axis, but we have a non-constant signal in x direction, namely a one pixel wide peak at the x position of the line. So we expect that all y-axis frequencies in the Fourier transformed image to be zero, this means that only at y frequency zero we have non-zero values which describe the periodicity in x direction.
As we can see from figure 10 the transformation of a single line results into a single line in the frequency plane. The line in the frequency plane is perpendicular to the line in the image plane and is located in the center. This confirms our expectations.
Since we plot our results with a logarithmic scale, we need to keep in mind that $\log(0) = -\infty$. In order to be able to plot our results we added a small value ϵ to our Fourier transformed results just before plotting.



(a)



(b)

Figure 10: The image of a vertical line (a) and of a horizontal line (b) (images on the left side) are transformed to the frequency plane (images on the right side).

To explore the frequency plane further we plot the intensity at y frequency equals zero along the x frequency axis (frequency plane of image shown in figure 10 (a)). This describes the periodicity of the image in horizontal direction. As we see in figure 11 the intensity along this axis is constant.

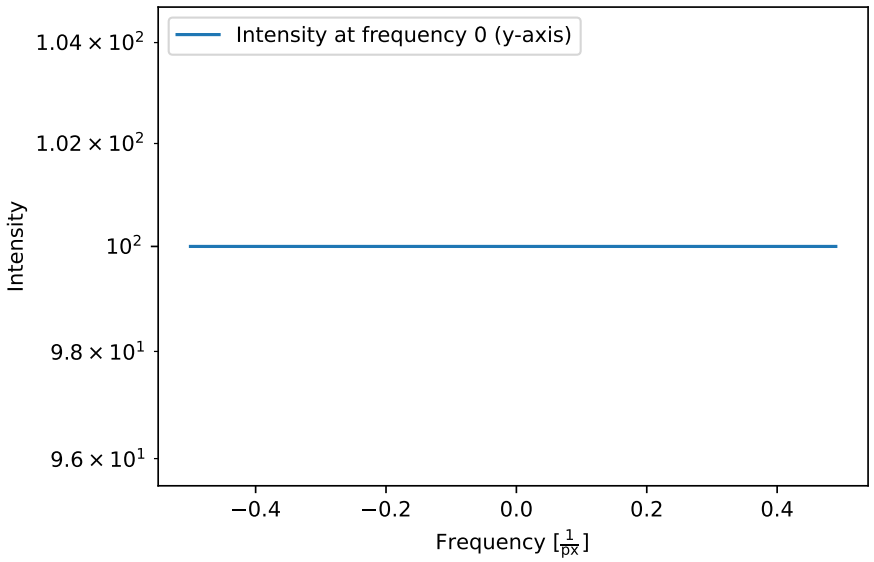


Figure 11: The intensity of the Fourier transformed image in figure 10 (a) at y frequency equals zero.

As a next step we want to find out what happens, if we insert a periodic signal into the image plane in the form of equally spaced lines. Figure 12 shows an image where there are several vertical lines with a spacing of 20 pixels. As in the case of the single line the image is constant in y direction and so all vertical frequencies are assigned zero. In x direction the lines create a periodic signal with frequency $\frac{1}{20} = 0.05\frac{1}{\text{px}}$. On the right side of figure 12 we see the fourier transformation of the image with the equally spaced lines and figure 13 shows a horizontal cut through the center. We see that only the pixels at $0.05n\frac{1}{\text{px}}$ $\forall n \in \{0, 1, 2, ...\}$ are non-zero.

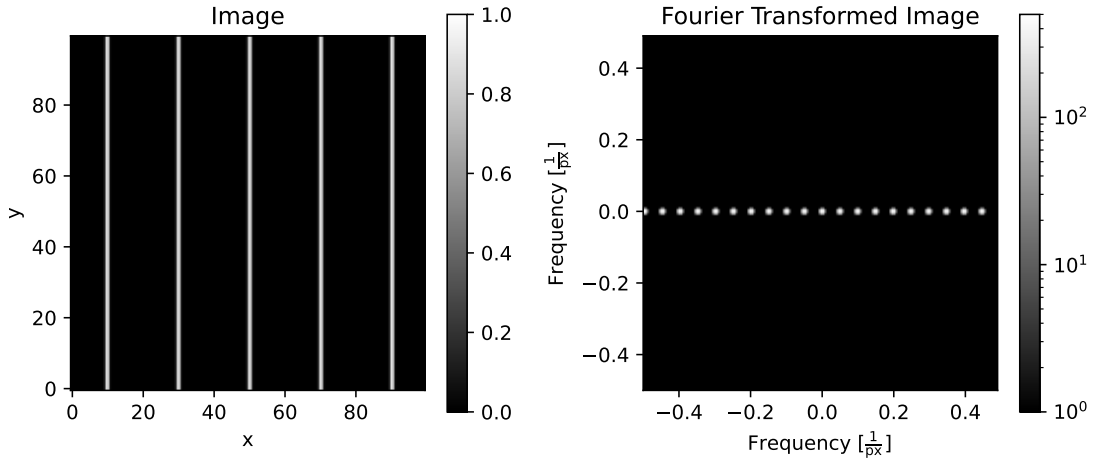


Figure 12: The image with several equally spaced one pixel thick lines on the left is transformed to the frequency plane, see image on the right.

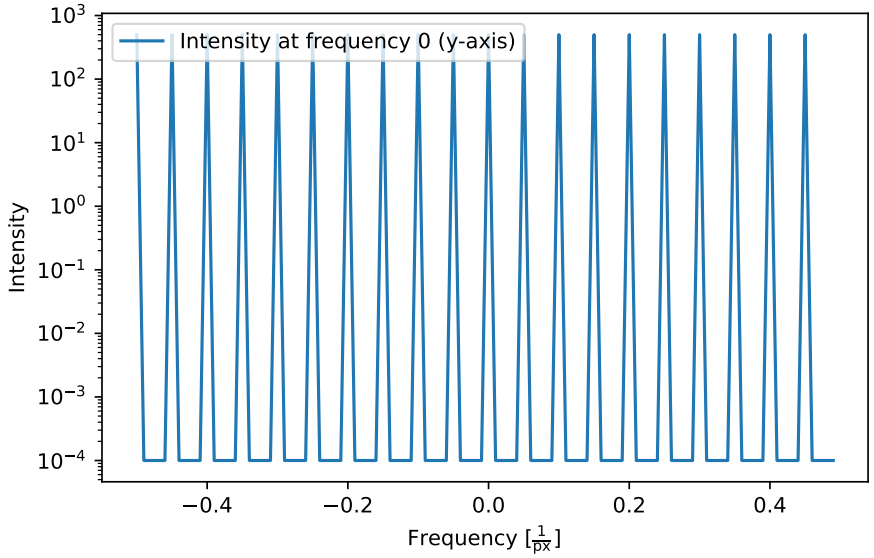


Figure 13: The intensity of the Fourier transformed image in figure 12 at y frequency zero.

In the appendix A we show, what happens if one line in the periodic image is missing and thus the signal is not completely periodic. We find that the treshold raises up to a higher value.
The spiders are not lines, but they have also an expansion into the horizontal, so we are also interested to see the Fourier transform of a single beam. We investigate the image of a beam (stair function) with a width of 10 pixels, placed at $x = 10$. Figure 14 shows the corresponding image and its Fourier transform. As with the lines the only frequencies in the frequency plane with a non-zero value are along the $y = 0\frac{1}{\text{px}}$ frequency axis. Figure 15 shows this axis in more detail. In contrast to the frequency plane of the line we have a signal which is stronger for central frequencies and decreases slightly for larger frequencies. Additionally we have strong minima at $0.1n \ \forall n \in \mathbb{N}$, where the position of the minima is given by one over the width of the beam.

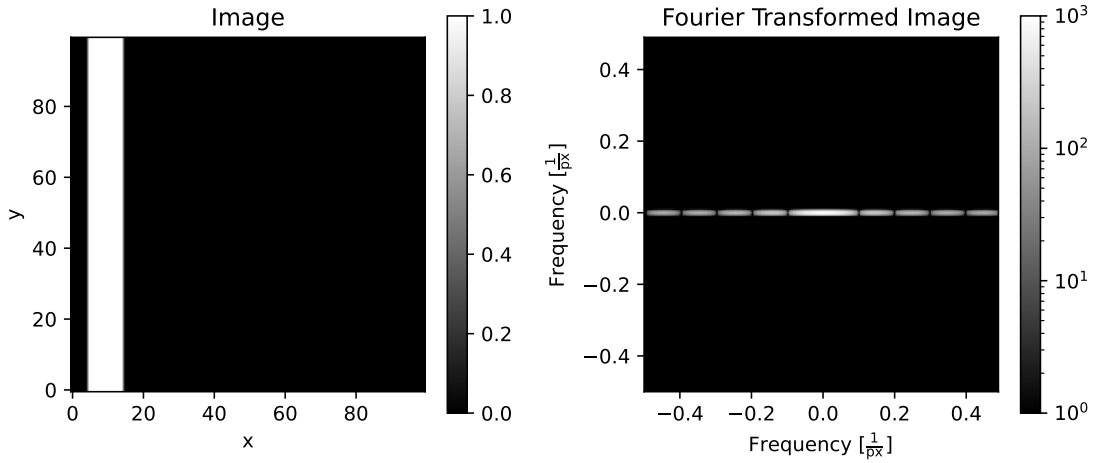


Figure 14: The image of one 10 pixel thick beam on the left is transformed to the frequency plane, see image on the right.

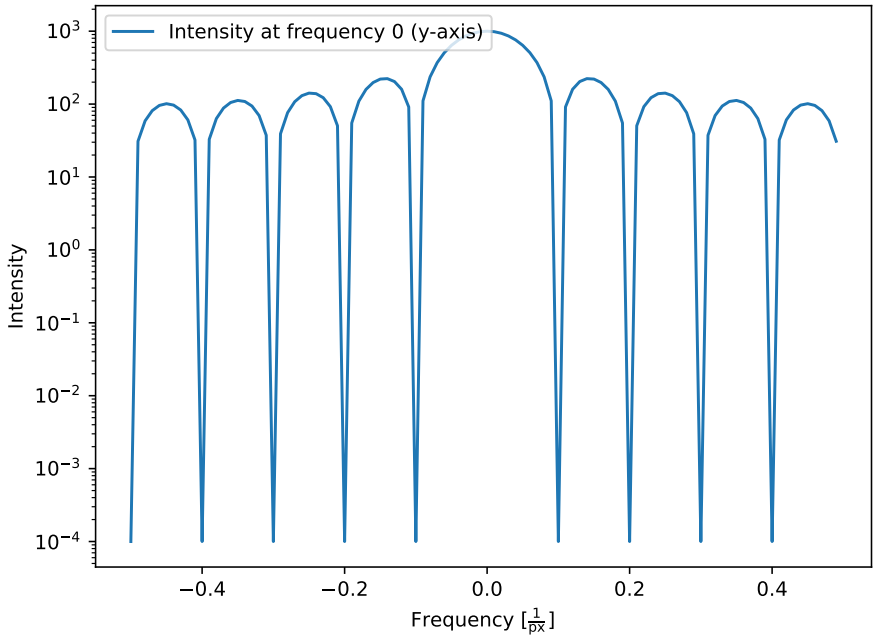


Figure 15: The intensity of the Fourier transformed image in figure 14 at y frequency zero.

As before with the lines we have a look at what happens if we have several of this beams, again equally spaced with a spacing of 20 pixels. We find that this image, lets call it $h(x)$, is a convolution of the image with the equally spaced lines $f(x)$ and the image with the single beam $g(x)$, namely

$$h(x) = (f * g)(x) = \int_{\mathbb{R}^n} f(\tau)g(x-\tau)\mathrm{d}\tau. \tag{10}$$

From the convolution theorem we find that for the Fourier transform it yields:

$$\mathfrak{F}\{h(x)\} = \mathfrak{F}\{(f * g)(x)\} = (G \cdot F)(\mu), \tag{11}$$

where $G(\mu)$ and $F(\mu)$ are the Fourier transforms of $g(x)$ and $f(x)$ [4]. This means that the Fourier transform of the image with the several beams is given by the multiplication of the Fourier transform of the image with the equally spaced lines and the image with the single beam, which can be seen in figure 17. Around the center frequency we have some peaks separated by $0.05\frac{1}{\text{px}} = \frac{1}{20}\frac{1}{\text{px}}$ which describes the separation between the beams of 20 pixels. This peaks are surrounded by other peaks which are separated by $0.1\frac{1}{\text{px}} = \frac{1}{10}\frac{1}{\text{px}}$ which describes the width of the beams of 10 pixels.

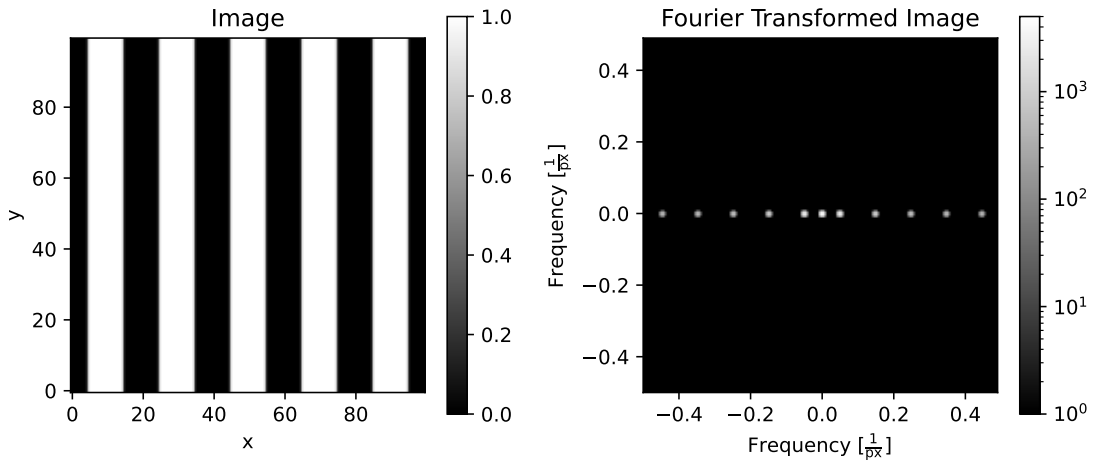


Figure 16: The image of several equally spaced 10 pixel thick beam on the left is transformed to the frequency plane, see image on the right.

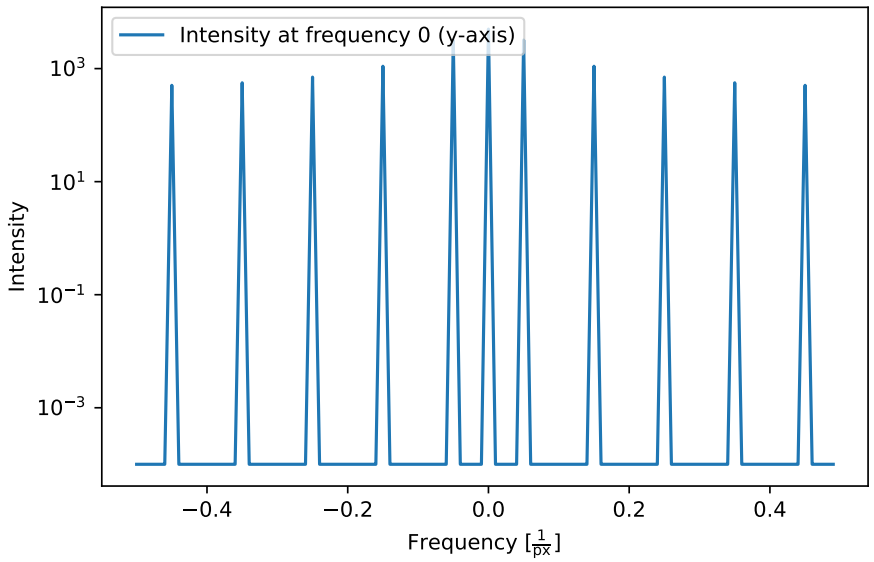


Figure 17: The intensity of the Fourier transformed image in figure 14 at y frequency zero.

5.2 Gaussian Beams

Before we have looked at beams with a stair function shape, but the spiders in our data do not have this stair function shape. In order to have a more realistic approximation we assume the spiders in our simulation to be Gaussian along the radial direction. We calculate the Gaussian profile from

$$f(x) = \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{12}$$

where μ is the mean (location) and σ is the standard deviation (width).
Also we change to the image format of the warped image which is not quadratic but rectangular.

Four our simulations we choose the radius range 254 to 454 pixels and compare it to the data from HD142527 in this range, which can be seen in figure 18. We have chosen the radius range such that the ghosts lie within it and one of the ghost is in the center of the radius range. So that latter on we can make sure, that we are able to suppress the signal from the spiders without loosing much of the ghosts which we use to simulate a really bright exoplanet.

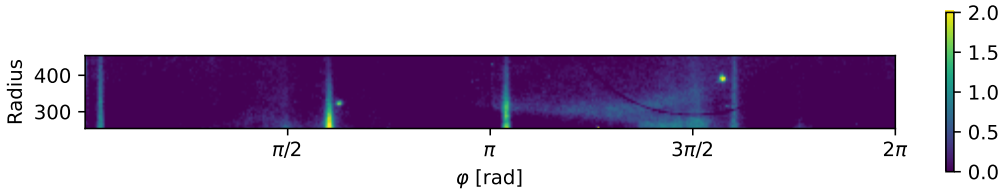


Figure 18: An image of HD142527 which is warped to the r-φ plane and the radial intensity drop-off is subtracted.

Figure 19 shows one of the spiders in a cutout from the image. We can see from the figure that a Gaussian is a good approximation for the shape of the spiders in radial direction.

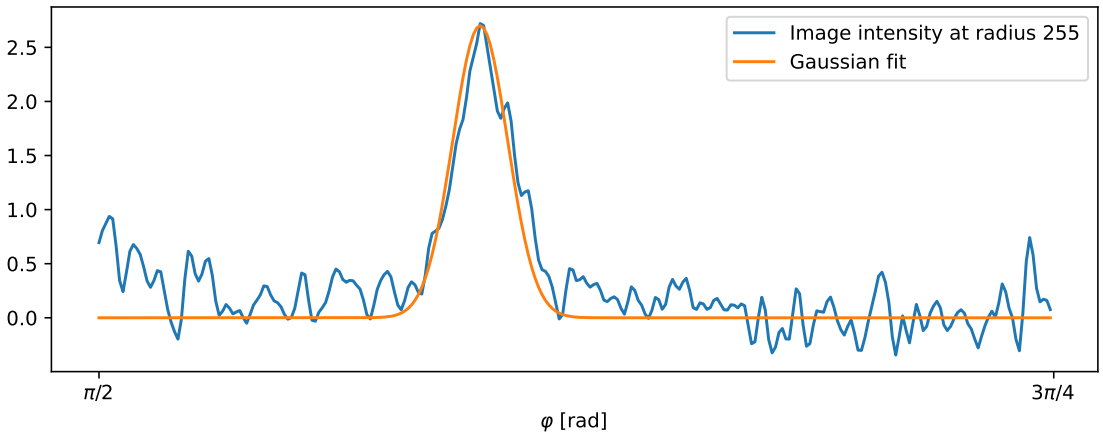


Figure 19: A cutout of the image intensity of HD142527 at radius 255, where we see one of the spiders and a Gaussian profile at the same position as the spider is. We can see that the Gaussian is a good approximation for the shape of the spider.

We know that the Fourier transformation of a Gaussian profile is again a Gaussian profile and from the previous subsection we know that the Fourier transform does not depend on the location from the beam, but on the width of it. The spiders in our data all have different widths. In figure 20 we plot four Gaussian profiles with different widths and their Fourier transforms. We see that indeed the Fourier transform of a Gaussian profile is also Gaussian (keep in mind that the plot is logarithmic) and that the width of the Gaussian profile is inverse proportional to the width of its Fourier transform as it already was the case for the stair function beams. Namely that the width of the Fourier transformed Gaussian is $w_{FFT} = \frac{1}{\sigma}\frac{1}{\mathrm{px}}$, where w_{FFT} marks the x-position where the Fourier transformed is again zero. Also the intensity of the Fourier transform decreases with the width of the Gaussian profile, however this effect is rather small in the logarithmic plot.

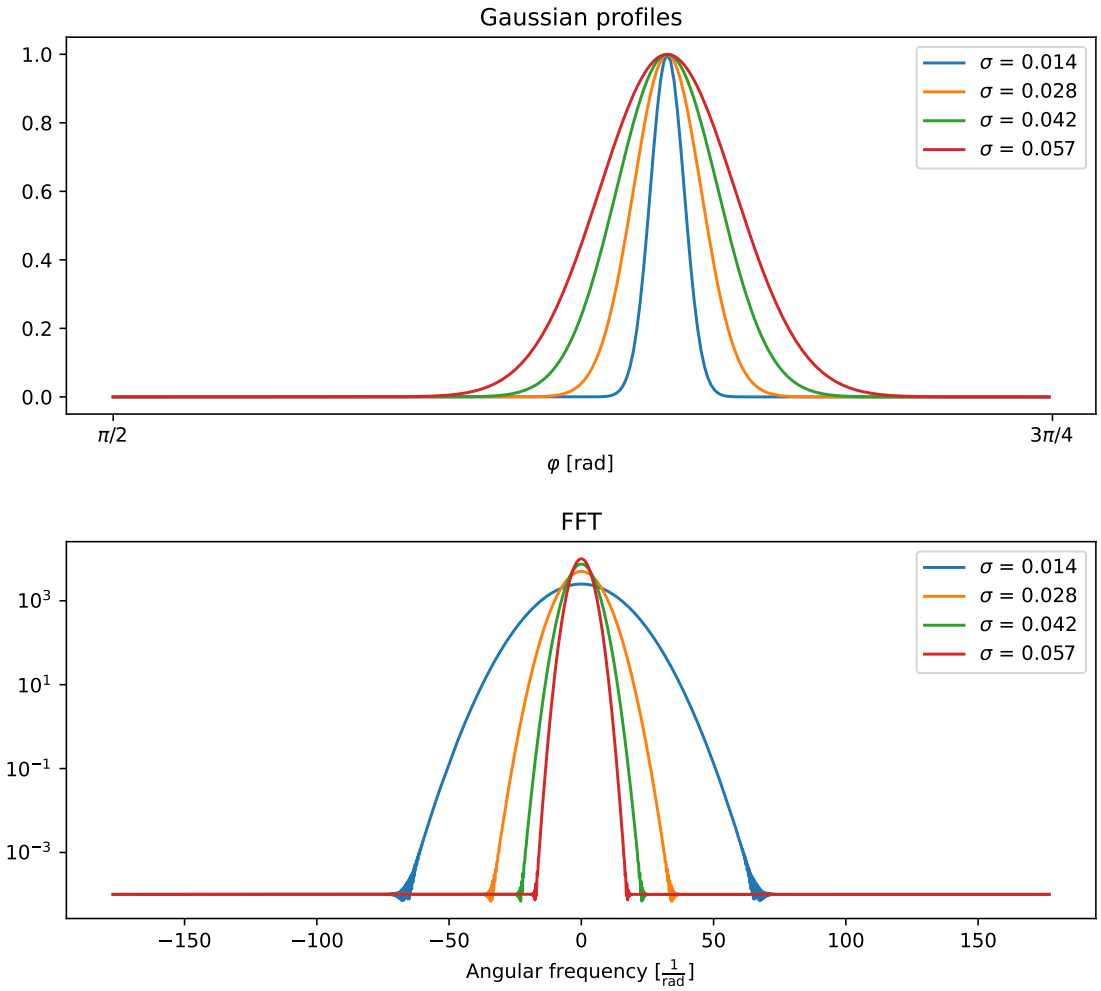


Figure 20: Gaussian profiles with different widths (top) and the respective Fourier transforms (bottom).

The VLT telescope produces four spiders which have a 180 degrees symmetry [5]. The spiders always have the same distance between each other, but the angular position and width can change from image to image. As a first approximation to this we have a look at four equal Gaussian beams which have the same spacing as in the data and fulfill the 180 degrees symmetry. Figure 21 shows a horizontal cut through this setup and the respective frequency plane. We also plot the Fourier transform of a single Gaussian profile. Due to the linearity of the Fourier transformation we expect the Fourier transformation of the four equal Gaussian to be four times the Fourier transform of the single Gaussian profile. As we see from figure 21, where also a Fourier transformation of a single Gaussian profile (green line) is contained, this is also the case, but there is a strong oscillation and a beat as well. To understand this oscillation we can use the knowledge which we have gained when we examined the behavior of the beams with stair function shape. There we found that if the image is a convolution of two different functions, then the Fourier transform is the multiplication of the Fourier transform of each function.
For a better understanding we have a look at four Gaussian beams which are all separated by 90 degrees, shown in figure 22. Here we have as one function the Gaussian profile with a width of 0.034rad, which if we Fourier transform it produces a Gaussian profile in the frequency plane with a width of $\frac{1}{0.034} = 29.497\frac{1}{\text{px}}$. The other function in the convolution consists of four lines separated

by $\frac{\pi}{2}$rad, which in the frequency plane corresponds to a large intensity every $\frac{1}{\pi/2} = 0.637\frac{1}{\text{px}}$ pixel. The multiplication of this two Fourier transformed function results exactly in the image we see in figure 22.
When we go back to the setting we have in our simulations of the spiders, we do not have equally separated Gaussian profiles. We have the 180 degrees symmetry plus an unknown separation between the neighboring spiders (or Gaussians), this additional information causes an additional convolution which produces the frequency plane we see in figure 21.

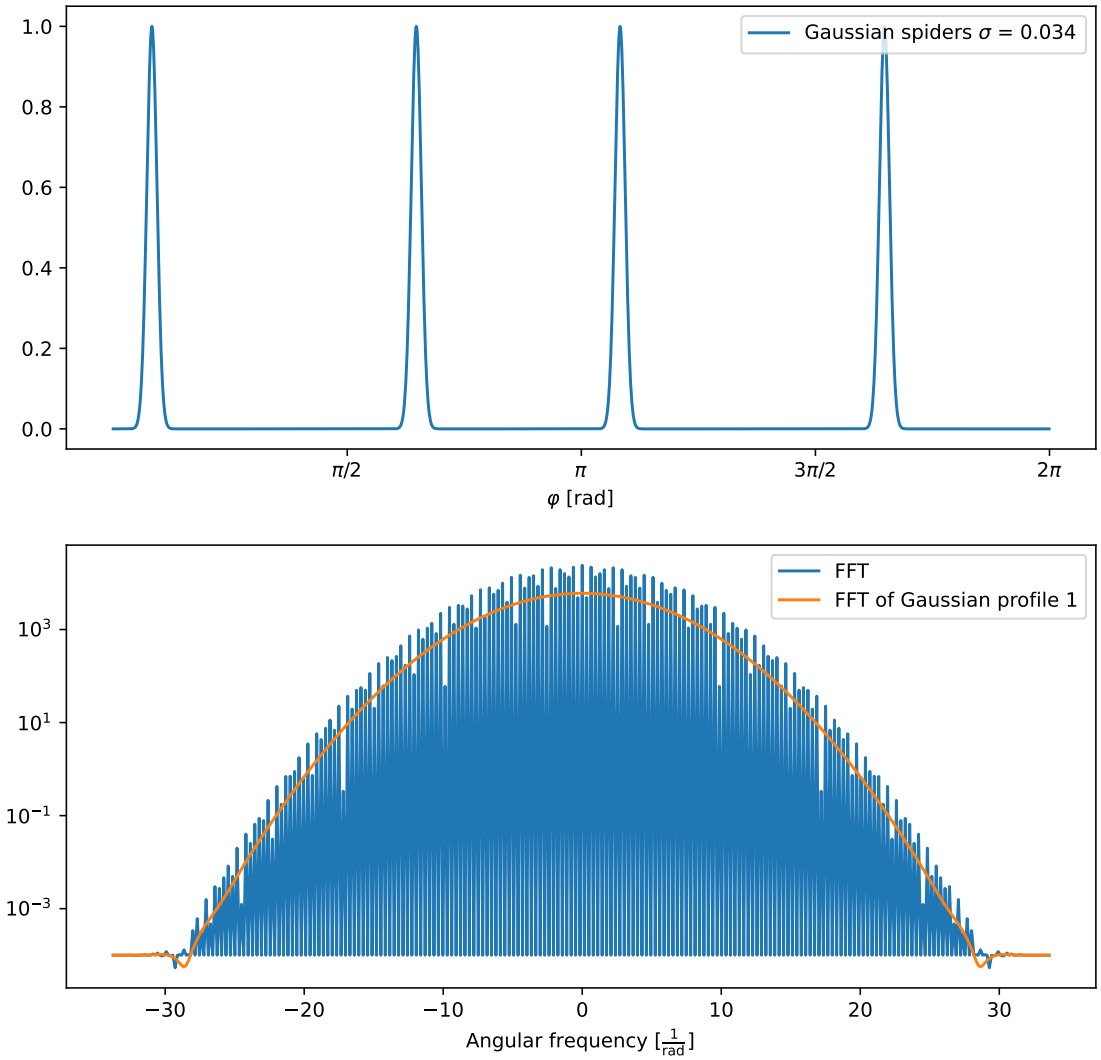


Figure 21: The top shows a function with four Gaussian profiles which have a 180 degrees symmetry. The Fourier transform of this is shown in blue in the bottom image. Additionally we plot in green the Fourier transform of a single Gaussian profile.

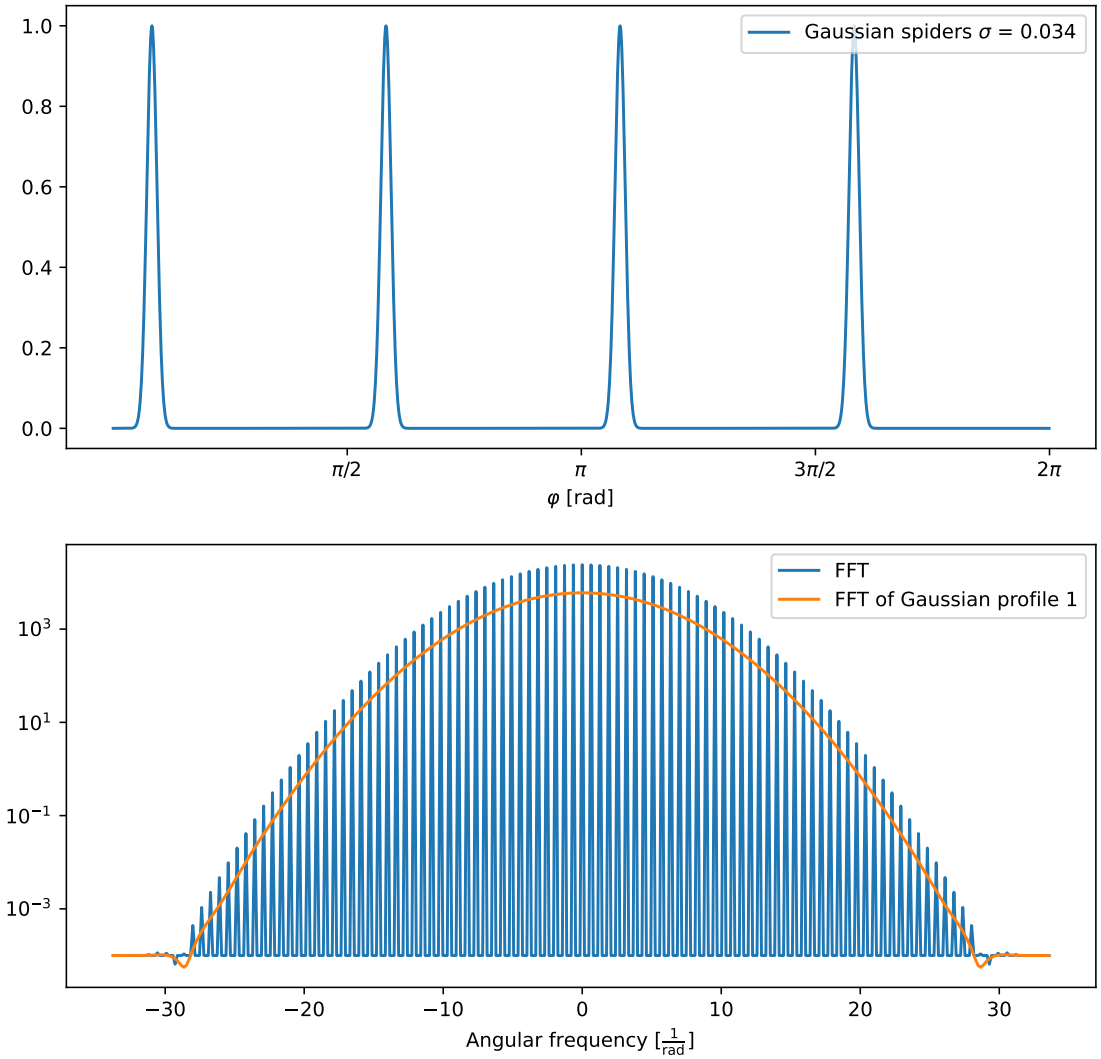


Figure 22: The top shows a function with four Gaussian profiles every 90 degrees. The Fourier transform of this is shown in blue in the bottom image. Additionally we plot in green the Fourier transform of a single Gaussian profile.

As a next step we want to include the fact, that the spiders have different widths, see figure 23. As we saw from figure 20 the width of the Gaussian mainly influences the width of the Fourier transformed Gaussian and its height. As before we can use the linearity of the Fourier transformation, but now we add up four different Gaussian profiles. Therefore the resulting Fourier transform has the width of the Fourier transform of the thinnest Gaussian profile and the height of the Fourier Transform from the thickest Gaussian profile. We see that the oscillation caused by the convolutions only starts to play a role when the intensity from the respective profile is large enough.

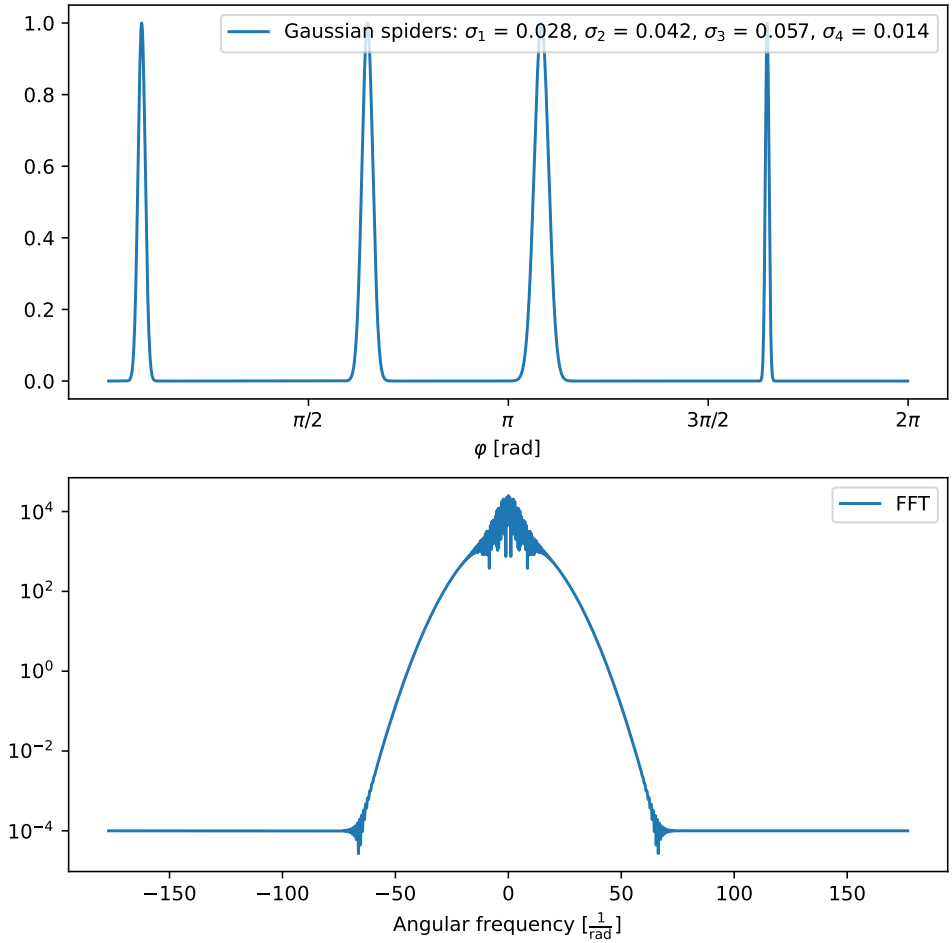


Figure 23: The top shows a function with four Gaussian profiles of different widths which have a 180 degrees symmetry. The Fourier transform of this is shown in the bottom image.

Finally we also need to take into account that the spiders have different intensities/heights. We look again at the four Gaussian profiles with equal widths which have a 180 degrees symmetry. This time the four Gaussian profiles have different heights as it is shown in the image at the top of figure 24. Figure 24 also shows the Fourier transform of each profile. We see that the Fourier transform of a profile with a higher intensity also has a higher intensity. When we Fourier transform the whole function which contains all four Gaussian profiles, we get the same Fourier transform as for the one with four equal Gaussian profiles, see figure 21, but the intensity of the oscillations is smaller and the intensity of the Fourier transform is the summation of the Fourier transform of each profile.

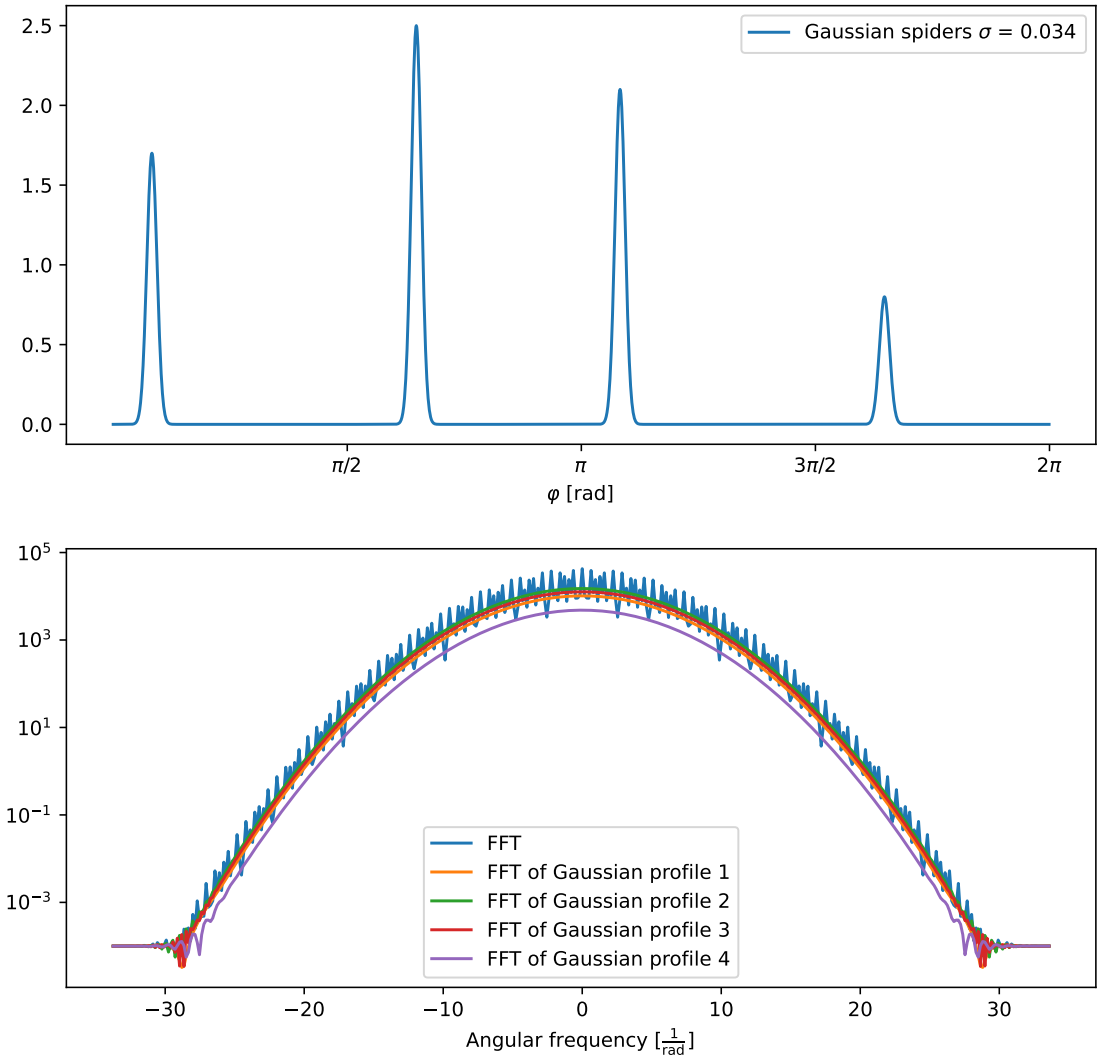


Figure 24: The top shows a function with four Gaussian profiles of different heights which have a 180 degrees symmetry. The Fourier transform of this is shown in the bottom image.

In conclusion we can say that for four spiders with a 180 degrees symmetry which have different widths and different heights the important frequencies are around the center frequencies. Also the strength of the oscillations are due to the different widths and heights not really strong, so that the intensity could also be approximated by the mean value to cancel out the oscillations.

5.3 Spiders

For the simulation of the spiders we use the knowledge which we have gained before from the Gaussian beams. In angular direction the spiders have a profile which is similar to a Gaussian, we therefore use Gaussian profiles for the simulation. In radial direction the spiders get thinner and their intensity decreases. Additionally all spiders have different widths and heights/intensities. Figure 25 shows the simulation of the spiders and the Fourier transformation of it. We see that in contrast to the Fourier transformation of the beams, Gaussian or stair function, there are not only non-zero frequencies at radial frequency zero, but they spread into all radial frequencies around the central angular frequencies. This radial spread is caused by the change of the width and intensity

in radial direction.

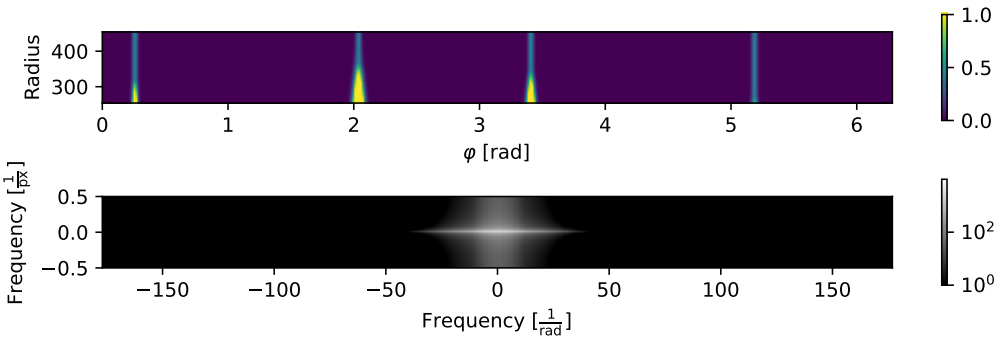
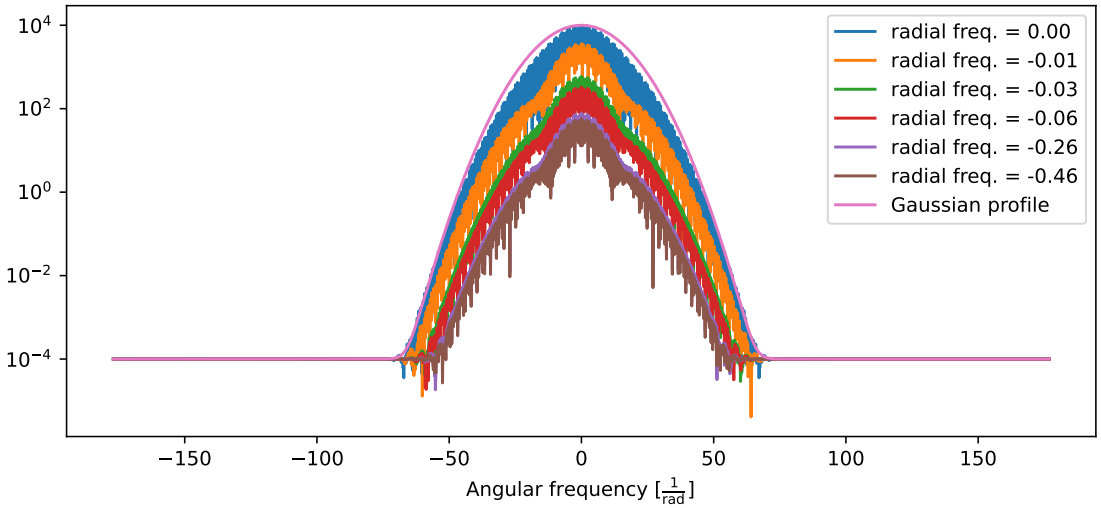
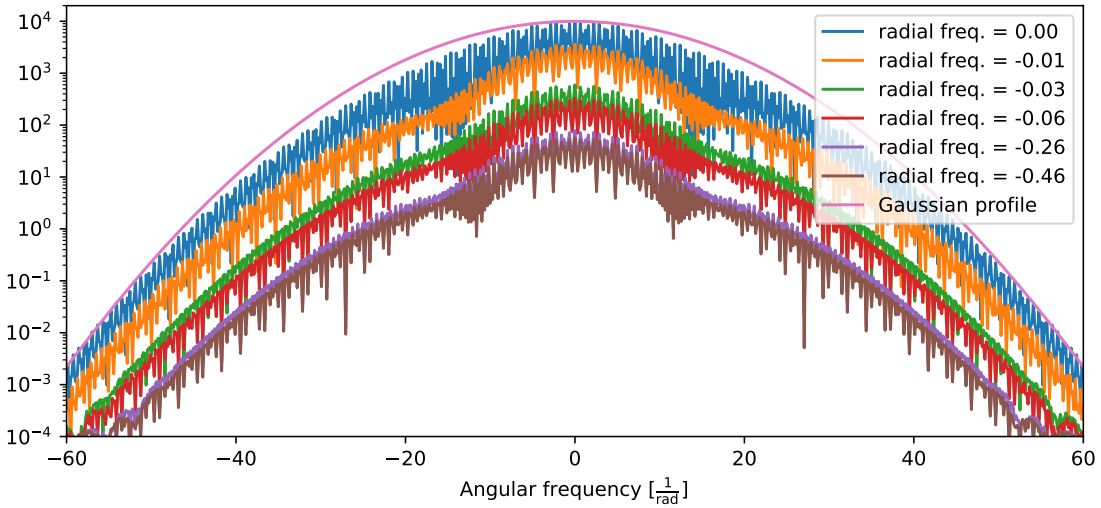


Figure 25: For a better understanding we simulate the spiders (top) and Fourier transform it (bottom).

For a better understanding of the Fourier transformation, we plot the Fourier transform at different radial and angular frequencies, see figure 26 and 27.
We find that the intensities at radial frequency zero are the same as if we the spiders would be Gaussian beams, with the intensity and widths being constant in radial direction. This means the width is defined by the inverse of the width of the smallest spider and the shape around the peak is dominated by the Fourier transform of the thickest spider. The maximal intensity is a sum of all Fourier transformed Gaussian profiles and the oscillations are due to the distances between the spiders.
For radial frequencies other than zero the intensity decreases rapidly to a small value and then stays constant, see figure 27. This can also be seen in figure 26, and therefore the important frequencies are for radial frequencies larger/smaller than approximately -0.6/0.6 $\frac{1}{\text{px}}$ and for angular frequencies larger/smaller than approximately -30/30 $\frac{1}{\text{rad}}$.



(a)



(b)

Figure 26: The Fourier transform from figure 25 at different angular frequencies and a Gaussian profile as reference (a) and an enlarged view of it (b).

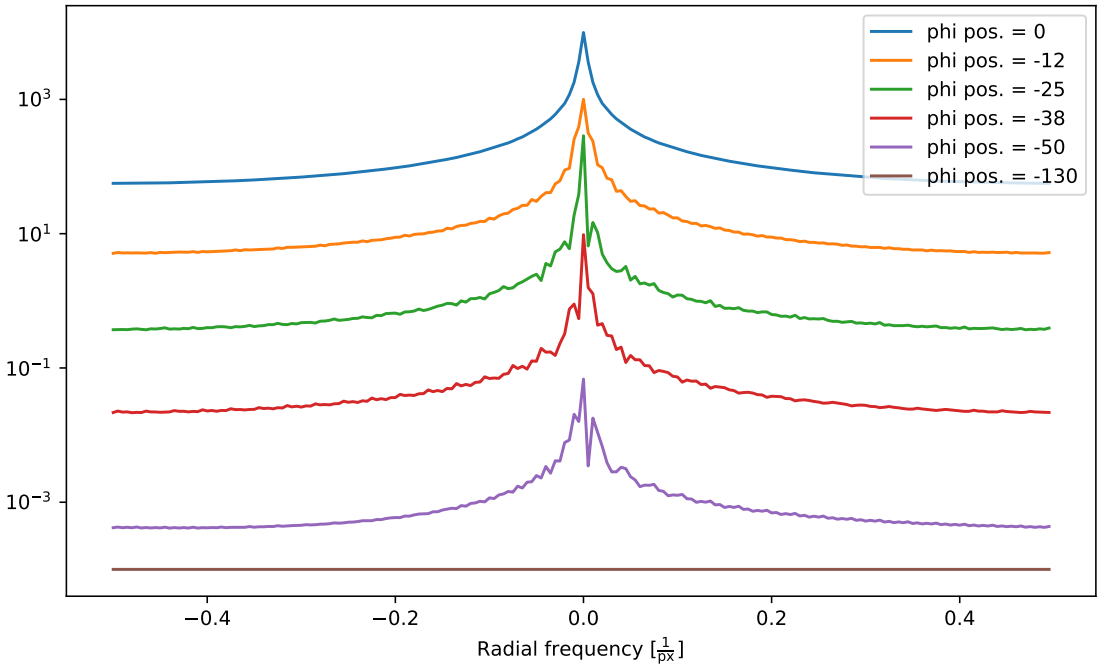


Figure 27: The Fourier transform from figure 27 at different radial frequencies.

5.4 Point-Like Sources

In order to make sure that we are not going to suppress the signal of point-like sources as exoplanets we need to know how their signal is going to look like in the Fourier space.
We investigate the signal of a Gaussian point source and of a point spread function (PSF). A PSF describes how a point source looks like after it has gone through an imaging system [1], which is in our case the VLT telescope with its adaptive optic. In order to simulate the PSF we used the python package AOtools [6].
Figure 28 shows a cut through the Gaussian and the PSF profile in the image plane. We want to compare the Fourier transformation of the PSF to the one of the Gaussian profile, of which we expect the Fourier transform to be again a Gaussian profile. The image with the PSF and its Fourier transform is shown in figure 29. Figure 30 shows the Fourier transform of a Gaussian profile and the PSF at radial frequency zero. As expected the Gaussian profile stays Gaussian in the frequency space. The Fourier transform of the PSF produces the same intensity for radial and angular frequency zero, but it decreases slower and has some local maxima. The two local maxima at each side of the global maximum produce a specific pattern which is completely different to what we get from the spiders and might be helpful to extract information about point sources from the frequency plane.

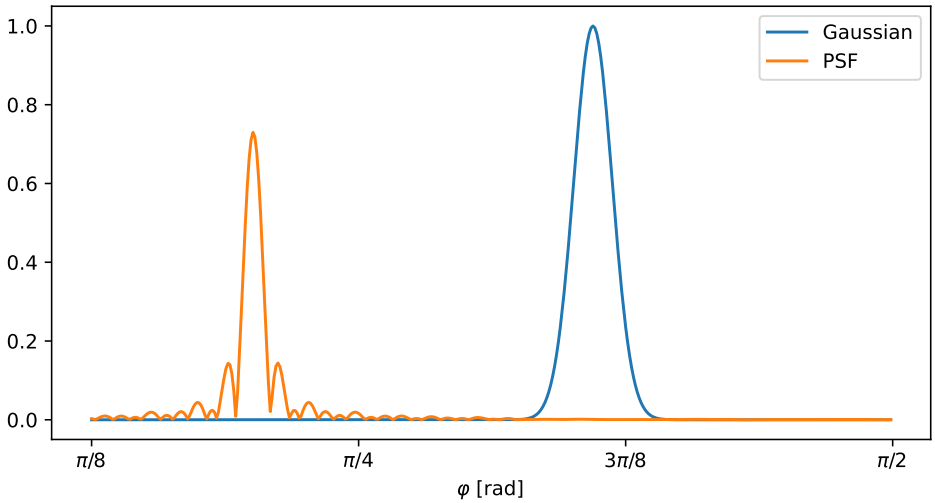


Figure 28: A cut through a Gaussian profile and a PSF along the angular direction. Both profiles are rotationally symmetric.

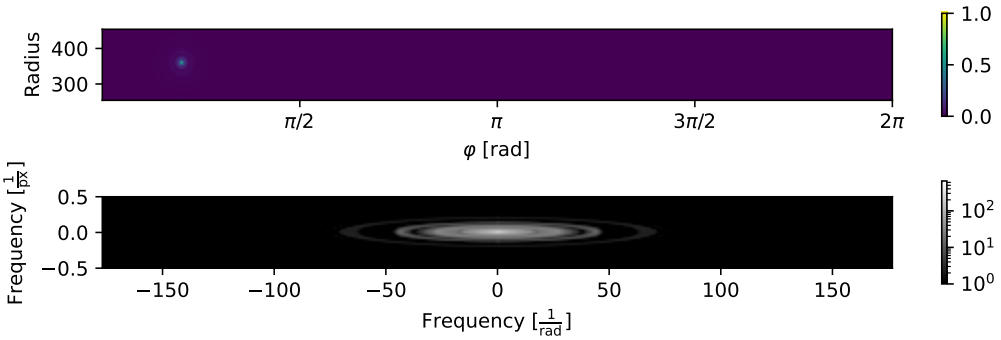


Figure 29: An image with a PSF (top) is transformed into the frequency space (bottom).

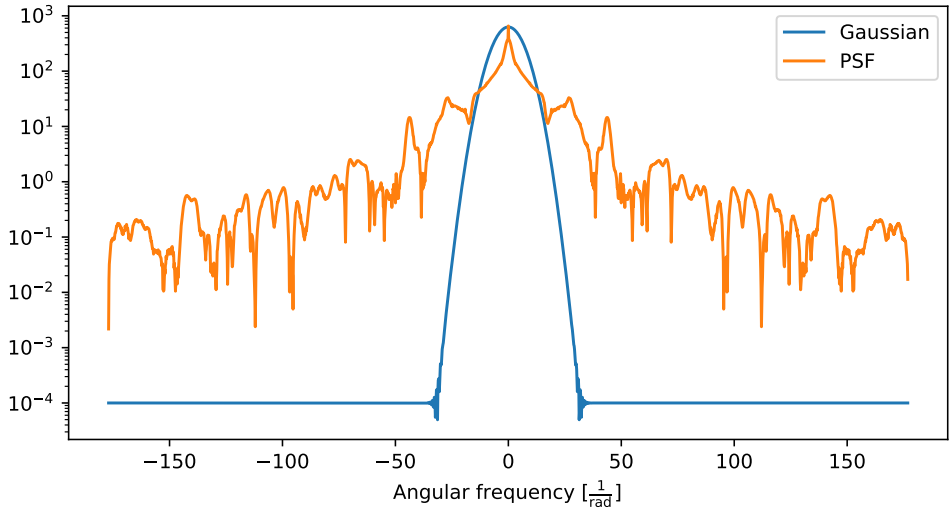


Figure 30: The Fourier transform of a Gaussian profile and a PSF at radial frequency zero.

6 Acknowledgments

A FFT of an almost Periodic Signal

In section 5.1 we have seen the Fourier transform of an image with equally spaced lines. This is a periodic pattern. But what happens if the pattern is not completely periodic anymore? By taking out one of the lines we explore the effect on the Fourier transform. In figure 31 we see the image, where the line which should be at $x = 90$ is missing. From figure 32 we see that the frequency spectrum still has the same shape, but the intensity range changes. The background which before was at zero (in order to plot logarithmic we added a small value) is now lifted to an intensity of 100.

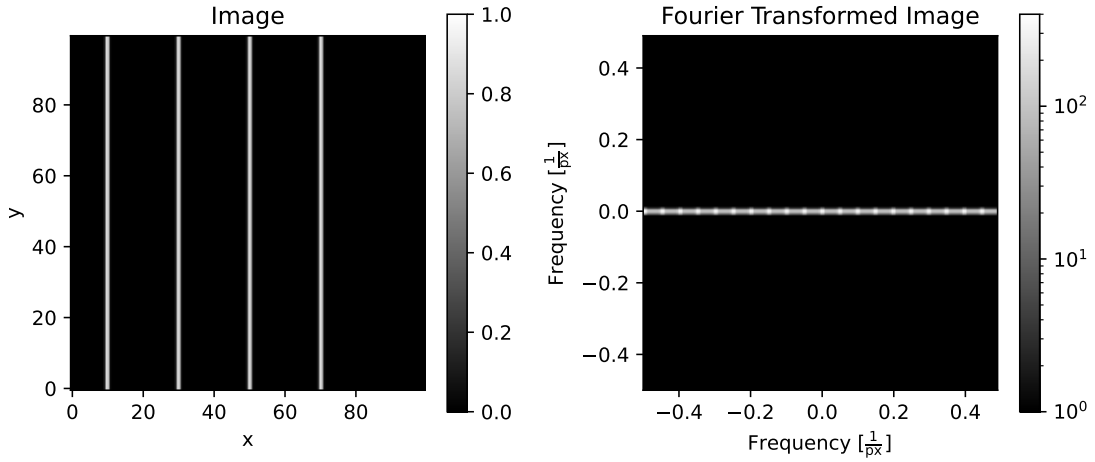


Figure 31: An image with several one pixel thick lines which are almost periodically distributed on the left is transformed to the frequency plane, see image on the right.

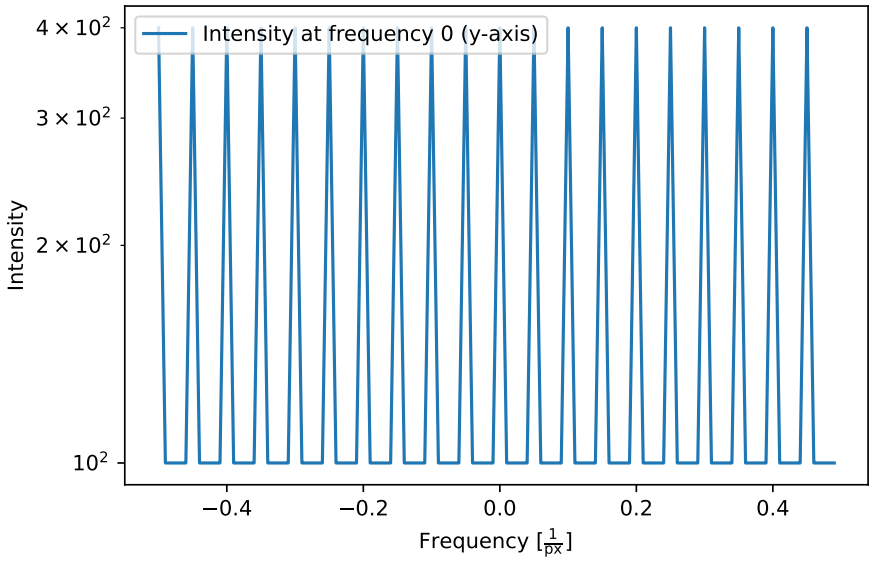


Figure 32: The intensity of the Fourier transformed image in figure 31 at y frequency zero.

References

[1] Point spread function - Wikipedia.

[2] Christian Bauckhage. Numpy / scipy recipes for image processing: General image warping. 01 2019.

[3] Gisin Dominique. HR 8799 imaged in the long I band. 2021.

[4] Rafael C. Gonzalez. *Digital image processing.* Pearson, New York, NY, fourth edition, global edition edition, 2018.

[5] L Schmidtobreick and E Pompei. EUROPEAN SOUTHERN OBSERVATORY Organisation Européenne pour des Recherches Astronomiques dans l'Hémisphère Austral Europäische Organisation für astronomische Forschung in der südlichen Hemisphäre VERY LARGE TELESCOPE NACO User Manual.

[6] M. J. Townson, O. J. D. Farley, G. Orban de Xivry, J. Osborn, and A. P. Reeves. Aotools: a python package for adaptive optics modelling and analysis. *Opt. Express*, 27(22):31316–31329, Oct 2019.