ARCHI: pipeline for light curve extraction of CHEOPS background stars

André M. Silva1,2*, Sérgio G. Sousa1, Nuno Santos1,2, Olivier D. S. Demangeon1, Pedro Silva1,2, S. Hoyer 3, P. Guterman 3,4, Magali Deleuil3, David Ehrenreich 5,

1. Instituto de Astrofísica e Ciências do Espaço, Universidade do Porto, CAUP, Rua das Estrelas, 4150-762 Porto, Portugal
2. Departamento de Física e Astronomia, Faculdade de Ciências, Universidade do Porto, Rua Campo Alegre, 4169-007 Porto, Portugal
3. Aix Marseille Univ, CNRS, LAM, Laboratoire d’Astrophysique de Marseille, Marseille, France
4. Division Technique INSU, BP 330, 83507 La Seyne cedex, France
5. Observatoire astronomique de l’Université de Genève, 51 chemin des Maillettes 1290 Versoix, Switzerland

E-mail: amiguel@astro.up.pt
† https://github.com/Kamuish/archi

ABSTRACT

High precision time series photometry from space is being used for a number of scientific cases. In this context, the recently launched CHEOPS (ESA) mission promises to bring 20 ppm precision over an exposure time of 6 hours, when targeting nearby bright stars, having in mind the detailed characterization of exoplanetary systems through transit measurements. However, the official CHEOPS (ESA) mission pipeline only provides photometry for the main target (the central star in the field). In order to explore the potential of CHEOPS photometry for all stars in the field, in this paper we present archi, an additional open-source pipeline module† to analyse the background stars present in the image. As archi uses the official Data Reduction Pipeline data as input, it is not meant to be used as independent tool to process raw CHEOPS data but, instead, to be used as an add-on to the official pipeline. We test archi using CHEOPS simulated images, and show that photometry of background stars in CHEOPS images is only slightly degraded (by a factor of 2 to 3) with respect to the main target. This opens a potential for the use of CHEOPS to produce photometric time series of several close-by targets at once, as well as to use different stars in the image to calibrate systematic errors. We also show one clear scientific application where the study of the companion light curve can be important for the understanding of the contamination on the main target.

Key words: techniques: photometric – techniques: image processing – methods: data analysis – stars: planetary systems

1 INTRODUCTION

The CHaracterizing ExOPlanet Satellite, CHEOPS (Broeg et al. 2013; Fortier et al. 2014; Rando et al. 2018), the recently launched ESA’s mission is the first dedicated mission to better characterize planetary transits with ultra-high precision photometry on stars known to have exoplanets.

The orbit during the mission operation is a circular Sun-synchronous orbit, at an altitude of 800 km and an orbital period of approximately 100 minutes. The spacecraft is nadir locked and thus it will always be rotating around Earth, pointing towards the targeted direction. It has a 1024x1024 pixel Charge Coupled Device (CCD) but only a smaller window with a default size of 200x200 pixel is sent back to Earth. This region, or SubArray, is free to be centered anywhere on the CCD, allowing the possibility to select better regions of the full ccd for more precise photometry. The orbital configuration of the spacecraft results in the rotation of the field of view and, consequently, the background stars rotate over the CCD, at a rate of 3.6 deg/min. The satellite exhibits a small field of view, with a diameter of 0.32°, a pixel scale of “1” and a Point Spread Function (PSF) with a radius of 12 pixels Hoyer et al. (2019). It is expected that the targets have magnitudes in the 6 to 12 V-mag range, with an exposure time that can changed between 1 ms to 60s, depending on the target’s brightness.

This rotational movement will lead to the background
stars occupying different pixels and, due to this constant change, we expect to find lower photometric precision when compared to the target star, which is maintained centered in a specific region of the CCD. On top of this, the exposure time is set taking into account the brightness of the central star, leaving the background ones in sub-optimal conditions, i.e. under or over exposed, either not being able to be detected or are saturated. Nevertheless, there will be cases where the background stars observations can be still used to extract precise light curves, as we shall see later in this paper.

In line with the main goals of the CHEOPS mission, the official Data Reduction Pipeline (Hoyer et al. 2019), henceforth DRP, only provides photometry for the main target, i.e. the central star in the field. However in the same field of view we can find other stars, for which it may be scientifically useful to derive (for free) precise photometry timeseries. Precise photometry of the background stars can be used to diagnose systematic errors in the data and possibly allow to correct for spurious signals in the photometric timeseries of the main target, such as contamination from background stars, as shown in Section 4.4. If sufficient precision is achieved, this can in principle be used to confirm the astrophysical origin for observed signals. Examples include the contamination by background bright eclipsing binaries (Abdul-Masih et al. 2016; Deleuil, M. et al. 2018) and to diagnose other unexpected effects, including instrumentation-related. Finally, since 20% of the available observing time of CHEOPS is open to the community (Guest Observing), other science cases, not related with exoplanet science, may gain from the observation of several stars in "dense" fields. Examples include photometry of open cluster stars that can’t at present be analyzed using the official DRP. It is thus important to access the photometric precision that can be achieved for background stars in the CHEOPS field. This is the main goal of the present paper.

We will start by describing the DRP in Section 2. Afterwards, in Section 3 we shall present the algorithms that were implemented in archi, our expansion to the DRP. Lastly, in Section 4 we shall see how archi behaves under different observational conditions and benchmark the different routines.

2 THE OFFICIAL DATA REDUCTION PIPELINE (DRP)

Taking into account that this pipeline was built as an extension for the CHEOPS mission DRP, we must understand what this can do. As the pipeline is introduced in Hoyer et al. (2019), we shall only give a brief introduction to the parts relevant to our work.

In a generalized way, we can describe the DRP as a collection of 3 different modules, that are applied in a sequential order, as presented below:

(i) **Calibration**: Corrects the instrumental response, by attempting to remove:
- Bias and readout noise;
- Non-linearity of the CCD;
- Gain;
- Dark current;
- Flat Field.

(ii) **Correction**: Corrects environmental effects, such as:
- Smear correction;
- Detection of bad pixels;
- Detection and correction of cosmic rays;
- Background.

(iii) **Photometry**: Extracts the targeted star’s light curve from the images.

The background estimation within the DRP (pipeline) is performed with an histogram based method, that uses an annulus centered on the target, not counting the pixels that are in the aperture. The background value is then estimated by fitting a fitted skewed Gaussian and, afterwards, used to correct the full images. For further detail we refer to Section 5.3 of Hoyer et al. (2019). We can now look into the Photometry module with greater detail, to understand the methodologies in use. The flux from the target star is calculated with a circular mask and, to maintain the same number of pixels in the mask, it is created once and afterwards it is shifted with an anti aliasing shifting algorithm, described below, to avoid altering the mask’s surface. This pipeline, extracts the photometry of the target star with four different apertures:
- **DEFAULT**: Uses a circular mask with a default radius of 32 pixels;
- **OPTIMAL**: The mask aperture is optimized through the maximization of the signal to noise ratio;
- **RINF**: The mask has a radius equal to 80 % of the default mask size;
- **RSUP**: The mask has a radius equal to 120 % of the default mask size.

To always have the mask centered over the central star, an iterative Gaussian apodization method is used to estimate the star’s position in each image. This algorithm begins by placing a mask on the image position estimated by on-board software and removing from the images the contributions from nearby stars and noise caused by the jitter. Afterwards, the centre of light, from this corrected image, is calculated, and a new mask is centered in it. This process then repeats until a convergence criterion is met, which tends to occur fairly quick, under 20 iterations. With this, it is capable of estimating positions with errors as low as 2x10⁻³ pix, as reported in (Hoyer et al. 2019). Due to those low errors we shall use those positions as one of archi’s star tracking methods, as we shall see in Section 3.2.

Lastly, since the mission is yet to see its first light, all obtained data sets are simulated, using CHEOPSim (Futyan et al. 2020), the official simulation tool.

3 METHODS

3.1 Finding the stars

Within this work we have developed two different methods to identify a star in the images. One of them makes use of the sky coordinates of the stars, the plate scale of the CCD and the rotation angle of the satellite to estimate positions within the image, Section 3.1.1. The other one applies image processing techniques to extract the star’s contour and it’s location in the image, Section 3.1.2. The former shall be referred to as the “fits” initial detection method, whilst the later is the “dynam” initial detection method.
3.1.1 Usage of a Star Catalogue

For CHEOPS observations we need to define a star catalogue, available with the DRP processed data, from which we can extract the right ascension (RA) and declination (DEC) of each star. Furthermore, from the DRP’s outputs, we know the scale of the images and the rotation angle of the satellite for each image. If we calculate the difference of RA and DEC between the target star and each non-target star and convert it to pixels, we know how far away each star is from the center.

Now that we know the separation, in pixels, we still need to take into account one last detail: the RA and DEC were calculated for a CCD with a roll angle of zero, which is not guaranteed to happen on the first image of the data set. Thus, we have to rotate the estimated points by an angle \( \theta \) given by equation (2), so that the stars are represented by a shape estimation that was not as sensible to those problems, but an interested reader can refer to Rocha et al. (2002).

The star catalogue contains many stars, most of which are very faint stars and end up not being actually detected in most CHEOPS observations, as we can see in Fig. 1. Those stars cannot be reliably studied with this pipeline and thus they must be removed from the pool of possible star positions. This can be accomplished with a simple magnitude filter, i.e. removing the points from stars with a magnitude higher than 13, as illustrated in Fig. 2, which is a very rudimentary way of accomplishing it. Since in CHEOPS the exposure time depends on the target star there is not a guarantee that this threshold is always valid, since with higher exposure times we could be able to detect stars with an higher magnitude. Thus, we were motivated to use a different approach that was not as sensible to those problems, with image processing techniques.

3.1.2 Usage of image processing techniques: contour detection

Due to the unique shape of CHEOPS’s Point Spread Function, or PSF (Hoyer et al. 2019), we attempted to track the stars by detecting them in the images. There are many approaches that one can use to track a moving object in images, but feature-based approaches tend to be more robust ones. This approach will be built around image moments, which we can think about as an weighted average of the pixel’s intensities that can be used to determine the image’s area, centroid and even some information on the image’s orientation. A further discussion is beyond the scope of this paper, but an interested reader can refer to Rocha et al. (2002).

Shape estimation Using both the zeroth, \( m_{00} \), and first degree, \( m_{01} \), moments we can estimate the centroid of any given shape, by applying equation (1), in which \( X_c \) and \( Y_c \) are the coordinates of the image’s centroid.

\[
X_c = \frac{m_{10}}{m_{00}} \\
Y_c = \frac{m_{01}}{m_{00}} 
\]  

(1)

Such algorithm can be easily implemented using Python’s OpenCV\(^1\) wrapper library, as seen in OpenCv (2019). However, before we can apply it, we have to perform some pre-processing steps to the images, so that they match the desired functions inputs. In order to avoid tampering with the photometric results, all changes are made over a copy of the original image that is only used to find the shape of the stars.

In order to properly use OpenCV we must convert our images to a suitable data type. We can convert them to various formats but, the easiest one to convert to, is from one in which each pixel has 16 bits, to one in which each pixel is an 8 bit unsigned integer. Since an 8 bit unsigned integer can only store numbers up to 255, we normalize the image in relation to its brightest point and then scale it up to 255.

Now that we have our image in the desired data type, we proceed to apply a binary threshold to the image, as given by equation (2), so that the stars are represented by a value of MaxValue and the background a value of zero, thus facilitating the next step in the process: finding the contours.

\[
I_{final}(x,y) = \begin{cases} 
\text{MaxValue}, & \text{if } I_{original}(x,y) > \text{threshold}. \\
0, & \text{otherwise.}
\end{cases} 
\]  

(2)

Both the contour detection and the moments calculation are, once again, handled by the OpenCV library. As a last step we have to take into account small positives, i.e. a small region of points that barely passed the threshold value and, consequently, its shape is detected. With this library framework, the contours are returned as a set of coordinates, specifying the 2D coordinates of each point in the contour. If we calculate the area of the PSF, assumed to have a radius

\(^1\) https://opencv.org/. Accessed: 28/8/2019.
of 12 pixels, we can roughly estimate that it should have ~452 pixels in it. Thus, if we discard the masks that have less than 50 points inside them, we can avoid the false positives created by noise or other artifacts on the image, as we do not expect to be able to detect and track (in a reliable way) such small objects.

If we have one star much brighter than the others due to one of the background stars having an higher magnitude than the target, we might run into problems with the aforementioned normalization routine. As each image is normalized to its maximum flux value, the saturation of one of the stars might push other, fainter, stars below the binary threshold value. To circumvent such problems, we perform an iterative normalization process where, after finding the stars, the N brightest ones, with N being set by the user, are removed and the analysis process is redone.

The binary threshold As we expect the detectable stars to be much brighter than the background level, we can set the threshold to a value of 10 which should be more than enough to detect them in the background of the normalized images. If we use a lower threshold we risk starting to include other noise sources in the images, which is not desired. On the other hand, the usage of this threshold is such that in cases where one of the stars is saturated, or much more brighter than the others, the normalization routine will make the faintest stars dip below the threshold value and thus not detected by the contour-detection algorithms, even though it can clearly be seen in the images, e.g. Fig. 7(d).

Overall characterization of the routine In Fig. 2 we see that when applying this technique, represented with the blue points, we retrieve initial detections very near the ones estimated with the Star Catalogue, in yellow.

Remembering from before, when estimating the position of each star with the Star Catalogue we found that there were some faint stars inside the image region. However, when using this method, we cannot find such faint stars, due to either using a image with less information due to the data type conversion, the fact that the normalization routine is made with the brightest star or maybe the stars are too faint to be reliably detected.

3.2 Tracking the stars

Due to the satellite’s rotation, the CCD is not always in the same orientation but, instead, it is also rotating around the target’s direction. To properly study the background stars, we need to be able to track them consistently between images. To accomplish it, we implemented and tested three different methods:

- “static”: In order to get the points from the i-th image, rotate the ones from the (i-1)-th image, by the rotation angle of the satellite;
- “offsets”: Subtract, from the background stars coordinates obtained with the “static” method, the jitter suffered by the central star, when compared against the first image. For the central star, we use the points estimated by the DRP.
- “dynam”: Tracks the stars using image processing, by applying the technique described in Section 3.1.2 to all images.

Even though all of the approaches are good enough to estimate centroids within the stars, the first two have too much jitter to be able to yield good light curves. Thus, for the background stars, the “dynam” method is the best option. Regarding the central star, both the “offsets” and the “dynam” method present somewhat similar results, depending on archi’s configuration, as we shall see later on, in Section 4.

3.2.1 Shifting the masks

Now that we know the position of the star in each frame, having the masks, Section 3.3, accompany the movements is trivial: we need to calculate the changes in both axis in relation to the initial position and, afterwards, shift the initial mask by that amount. This shift is truncated to the pixel size precision, however we can do a re-sampling of the image to increase it, as we shall discuss in Section 3.4.

With the masks moving, we may encounter cases in which the shift is such that part of the mask goes outside of the image boundaries, thus picking up “empty” space. Although the areas outside the image are typically small, they can still impact the overall quality of the light curve and introduce errors, mainly due to the fact that, in practice, a number of pixels “disappears” for that point in time. The mask breaching the image boundaries is expected to occur for stars close to the image’s edge, where either mask’s shape or small uncertainties in the tracking method can lead to this situation.

3.3 Photometric masks

Within this section we shall explore the different masks used for photometry. So far we have implemented two masks, a
mask: Simply adding a layer of 1 pixel around a mask, does not maintain its proportions. However, despite not being able to properly increase the mask whilst maintaining the original shape, this will allow us to have a finer control of the mask size during the optimization routine, where we attempt to find the optimal mask size.

3.3.3 Finding the optimal mask

After determining the initial masks, they are either equal to the star’s shape or a circle with a previously defined radius. However, we want to find the best mask for each star, i.e. the one that has the largest area of the star’s PSF, without picking up too much noise from the background. To do so, we change the mask’s size until we find the minimal value of the noise metric, Section 3.6. During this process, we expect the noise to decrease until it eventually hits a minimum value and, afterwards, starts increasing, as the mask starts covering a greater portion of the sky.

If during the optimization routine a mask finds itself occupying regions outside the image, we have a trade-off between the number of points for the light curve and the global noise in it. We gave preference to a higher number of points in the light curve to try to better assess the global behaviour instead of benefiting the precision in each point of smaller sampled light curve.

3.4 Background grid

Due to the usage of shape-based algorithms for the masks and star detection, alongside the integer conversions that must be made to convert positions into grid coordinates, the movements on the image are limited to the grid’s nodes. Thus, ideally, we would like to have an image with more resolution than the one that comes out of CHEOPS CCD.

When working with images, we do not have a continuous surface and, instead, we should think of it as a equally spaced grid, in which each node represents a pixel. Thus, if we wish to link the determined coordinates of a star with a given pixel in the image, we must make an approximation. Once again thinking on the pixel grid, all detections lie within the grid node 100 and 101, are considered to be in the same pixel. Consequently, the mask is placed in the pixel that corresponds to the integer part of the estimated location.

Re-sampling the images By increasing the number of nodes in our grid between any two pixels, we could approximate the coordinates to a node much closer to their actual value. To do so we can artificially re-sample the images, with a positive scaling factor, defined in equation (3), greater than one.

\[ \text{scaling factor} = \frac{N_{\text{increased}}}{N_{\text{original}}} \]  

3.3.2 Shape Mask

Once again making use of image processing techniques, we can extract the visible part of the star’s PSF, i.e. the line that delimits the star against the background, and use it to create a mask.

However, we now need to find a way to change the mask’s size, so it can be optimized. Unlike the circle mask, in which it’s straightforward to increase its size, we now need to find a way to increase it and to quantify such increase. To accomplish this, one can simply add layers of pixels around the shape, until the desired size is met. For example, an increase of 1, would add one layer of pixels around the entire mask, and so on.

In Fig. 3 we can see the method working, on a mask resembling a “plus” sign, adding a layer of 1 pixel around its edges. It’s also noticeable the flaw in the increase of the mask: Simply adding a layer of 1 pixel around a mask, does not maintain its proportions. However, despite not being able to properly increase the mask whilst maintaining the original shape, this will allow us to have a finer control of the mask size during the optimization routine, where we attempt to find the optimal mask size.

**Figure 3.** Shape increase method applied to a “plus” sign (stripped boxes), that was increased by one pixel.
To finalize, there is a detail that we must keep in mind: we can see the increase in the grid size, as an increase in the correlated points, i.e. if one pixel is transformed into 9, then those 9 pixels, before normalization, are equal to the first one, as well as all the errors associated with that pixel. To avoid introducing flux into the images, we normalize the block of pixels, so that its sum is equal to the original level. We confirmed conservation of the flux, up to machine precision, on the re-sampled grid.

3.5 Uncertainties

In the DRP the calculation of uncertainties in the light curve is made through Eq. 4:

\[
Err = \sqrt{\text{Flux} + \text{bg} + N_{\text{pix}} \text{N}_{\text{stack}} \text{gain} \times \text{ron}^2 + \text{dark} \times t_{\text{exp}} N_{\text{pix}}} \\
bg = \text{background} \times N_{\text{pix}} \times t_{\text{exp}}
\]  

where \(Err\) is the uncertainty for a given point, \(\text{Flux}\) is the flux of the corresponding point, \(N_{\text{pix}}\) the number of pixels inside the mask, \(N_{\text{stack}}\) is the number of stacked images, \(\text{gain}\) is the gain from the digital conversion process that occurs in the CCD, \(\text{ron}\) is the read out noise, \(\text{dark}\) is the dark current, \(t_{\text{exp}}\) is the exposure time for each image and, lastly, \(\text{background}\) is the flux from background objects. It’s important to note that this equation does not take into account the jitter of the spacecraft, since adding it is not trivial. In arch we shall also use this equation to estimate the uncertainties, although some care is needed.

First of all, the parameters stored in the DRP outputs are calculated over a 200x200 region, giving us a per-pixel value of each component, as we shall see in this Section. However, when using a background grid, the number of pixels inside the masks will be inflated, when compared to the cases in which the background grid is not in use. Thus, if we do not correct this inflation, we will have an overestimation of the uncertainties.

In order to correct this effect we have to convert the number of pixels from the increased grid to the normal one, which is accomplished with Eq. 5.

\[
\text{corrected size} = \frac{\text{total size}}{(\text{scaling factor})^2}
\]  

where \(\text{corrected size}\) is the corrected number of pixels inside the mask and \(\text{total size}\) the total number of pixels.

Another consideration that must be taken into account is that both the background and dark values are stored for the entire region of the DRP’s aperture, instead of being a per pixel value. Since the DRP’s mask is not a completely binary mask, as near the edges the pixels are weighted, it will not be possible to retrieve the exact values. As an approximation we shall consider the DRP’s mask to be a circular binary mask, thus allowing us to compute the per-pixel values, as we shall now see.

3.5.1 Background

The background stored in the DRP outputs is not the calculated value but, instead, it is re-scaled by mask’s size, as given by Eq. (6).

\[
\text{stored}_{\text{bg}} = \text{background} \times t_{\text{exp}} \times N_{\text{pix}}
\]  

With this knowledge, we want to extract the background per pixel, to calculate the uncertainties for our mask of choice. One aspect to take into account, would be the fact that the DRP’s background calculation is made over the image outside a region delimited around the target star, i.e. in the background value we can find contributions from the background stars. We could try to improve the background calculation with the masks and star tracking techniques so far described but, we would be introducing errors due to the movement throughout the CCD pixels and, some parts of the DRP would need to be re-implemented to allow us to work with images without background correction already applied on. Furthermore, as in the DRP Hoyer et al. (2019) the background correction routine already takes into account contaminating stars, we can use it to calculate the errors on our light curves. Thus, if we divide the \(\text{stored}_{\text{bg}}\) by the number of pixels inside the DRP’s mask, we get an estimation of the number of dark photons during the exposure time (actually exposure+reading time).

3.5.2 Dark

As CHEOPS has a small field of view and, consequently, the background stars are relatively close to the target, we can make a fair assumption that the dark value, estimated by the DRP for the central target, should be reliable also for these close by background stars. Furthermore, the typical dark current has a low value Hoyer et al. (2019).

Similarly to the background, the \(\text{dark}\) stored in the outputs is only calculated for the region near the central star, using the image outside the applied mask.

The actual values are stored in temporary files that are not available in the DRP’s processed data. Thus, to have this information, we would have to recalculate it. Even though the \(\text{dark}\) value is not stored, in DRP’s outputs we can find the dark component in the uncertainty calculation, given by equation (7).

\[
\text{stored}_{\text{dark}} = \text{dark} \times t_{\text{exp}} \times N_{\text{pix}}
\]

Since we know the radius of the circular mask used for our data set, we can simply divide this stored value by the number of pixels inside the mask, and thus have a rough estimation of the \(\text{dark}\). However, DRP makes use of 4 different masks and, if we are not careful, nearby stars can impact its value. Thus, in an attempt to minimize the contaminations, we can calculate the median of the estimated \(\text{dark}\) for each one of the 4 DRP’s apertures, and regard it as the \(\text{dark}\) value.

The downside of this method is that it assumes that we have an uniform dark, that is equal for both the target and the background stars, which may not hold as true. However, since the contribution from photon noise is a few orders of
magnitude higher than the one from the dark, those imperfections should not be very impactful. More so, the field of view is very small and therefore we expect that the dark will not change much in the image, specially in the subarray image, which is just a small piece of the CCD.

3.6 Noise metric - Combined Differential Photometric Precision

In order to estimate the noise in the light curves, an adaptation was made to the algorithm applied in NASA’s Kepler mission: Combined Differential Photometry Precision, CDPP, (Christiansen et al. 2012).

In Christiansen et al. (2012) words: “A CDPP of 20 ppm for 3-hr transit duration indicates that a 3-hr transit of depth 20 parts per million (ppm) would be expected to have a signal-to-noise ratio (S/N) of 1, and hence produce a signal of strength 1σ on average”, which is the ideal metric to quantify the noise existent on the light curve and, it can be interpreted as the effective noise seen by a transit pulse (Christiansen et al. 2012). In its calculation, the important factor is the near-term trend changes in brightness instead of the long-term ones (Koch et al. 2010).

For further details on the algorithm itself, one can refer to Jenkins et al. (2010); Christiansen et al. (2012), where the algorithm is properly introduced and discussed. However, for a brief introduction on the methodology behind it, I will now refer to one of its many adaptations, in Luger et al. (2016):

- Start by passing a 2 day quadratic Savitsky-Golay (Sav-Gol) filter to the flux, which should be more than enough to avoid fitting the transits in K2 Light Curves;
- Remove 5σ outliers;
- Divide the data into chunks, which have data from a time interval equal to the integration time. Afterwards the standard deviation is calculated for each one of them;
- Take the median of the standard deviations and divide by the square root of the number of points inside each chunk, obtaining the desired photometric precision.

The official DRP reports the usage of a modified CDPP algorithm, better described in Hoyer et al. (2019), which does not use any kind of detrending nor filtering within the metrics, also capable of taking into account eventual gaps in the data. The same algorithm was used in archi, to calculate the CDPP. To estimate the errors in this measurement, we built a 68% confidence interval of the standard deviation, across the chunks, with the 16th and 84th percentile. Thus, with this metric, we can evaluate the quality of the raw light curves.

4 RESULTS

Using the presented methods, we built archi: An expansion for the CHEops mission pipeline, an open source python package ², that works on top of the DRP outputs, allowing one to extract information from all of the stars present in CHEOPS observations. In the github repository one can find more information on how to use the library, alongside practical examples and, in Silva (2019), the pipeline architecture is thoroughly discussed.

Within this Section, we shall compare the different approaches and options in archi, i.e. the mask to be used, the size of the background grid, the initial detection method and the star tracking methods. This comparison will be made through the photometric precision, for a timescale of 30 minutes. As we still do not have real data from the mission, we made use of CHEOPS official simulator tool (Futyan et al. 2020) to create simulated data sets.

Simulated Datasets In order to study the behavior of archi we will use three different batches of simulated datasets. Data Set A has three stars rotating around the target star, all with different magnitudes. Data Set B1, B2 and B3 all have four stars, with the background ones having equal magnitudes in all three data sets. The only difference between them is the magnitude of the target star that is different to have different exposure times. Lastly, Data Set C has two visible stars rotating around the target star, all with the same magnitude. In, in Appendix A one can find more information on the different simulated datasets.

Naming conventions To simplify the linking of light curves to actual stars, each star is named using the following convention: “Star <index> (separation, ∆mag)” where index is an integer linked to the closeness to the centre of the image, separation is the distance from the target to the star and ∆mag is the difference of magnitudes between the star and target. Following this convention, the central star has index zero, the closest star to the target has index one, with the pattern being kept for all other stars. In Fig. (A1) we can see the name of each star, alongside their initial position in the image.

Lastly, from this point onwards, we shall set the convention of <mask type> - <initial detection> - <star tracking> to refer to the combination of methods that was used to obtain a given light curve.

4.1 Light curve extraction

Now that we have seen the different methods implemented in archi, we shall now characterize it. To do so, we shall start by looking at Data Set A, comparing the different combinations of star-tracking methods, presented in Section 3.

In the first place, we will see how the different combinations of masks and star tracking interact with each other. We have tested all combinations of initial detection methods and star tracking methods and decided to only show the ones that can achieve the best results and compare those against each other. Thus we shall only use the dynam initial detection method and the dynam and offsets star tracking methods.

Target star analysis

We can start by looking at the target star in Fig. 4 that, in a generalized way, has both masks yielding noise values close to each other, assuming that the same star tracking method is used. Even between both star tracking methods,
Table 1. Comparison of the CDPP for the light curve extracted, with archi and the DRP, from the target star, while using Data Set A.

| Pipeline       | Noise [ppm] |
|----------------|-------------|
| DRP            | 254±8.0     |
| archi - grid 0 | 192±6.2     |
| archi - grid 600 | 183±24.1  |

we fail to find big differences between the worst and best noise value. It’s also noteworthy that the increase of the background grid allows us to diminish the noise in the light curves. However, as one should expect, this improvement is limited, and grids bigger than 600, corresponding to a scaling factor of at least 3 times the original size of the image, do not reduce significantly the noise, despite the associated increase in the computational cost. Instead, from this point onwards, we find some small fluctuations in the noise level, which is a sign that we are close to the optimal conditions for the pipeline to work.

For this star, the error bars in the CDPP values stay relatively constant, without any visible changes neither between different methods nor with changes in the background grid in use.

Benchmarks against the DRP

Now that we have seen the performance of archi for the target star we can compare it against the DRP’s OPTIMAL light curve.

From Fig. 5 and Table 1 we see that the DRP produces a light curve with a lower flux value, albeit with an higher noise metric and a broader confidence interval. The higher error bars suggest that the DRP light curve is more affected by the systematics introduced by the rotation, since the points are more dispersed within each chunk used for the CDPP calculation.

In all comparisons made during this work, we have found that archi was consistently able to produce curves with a lower CDPP. Although, it’s important to note that the DRP is still under development and that this fact may not hold for later versions of it.

Background stars analysis

Contrasting what was found for the target star, when looking at the background stars, in Fig. 4, we find that the star tracking methods return results far apart from each other.

For the first two stars, Star 1 (5′′,+1) and Star 2 (7′′,-1), we see that both masks give similar results, with the main difference laying in the star tracking method. The dynam method reveals itself to be the superior alternative to track the rotating stars. If we now pay attention to the noise evolution with the background grid, we see that it stays almost constant for them all, without improving or worsening our results. The non-stationary nature of those stars translates into errors in the mask position and mask shape itself, which have not been counteracted by the increased resolution of the image. Within the background stars, we should also pay some attention to Star 3 (8′′,-0.5), the closest to the image’s border. In this one, the mask size is limited due to its closeness to the edge, thus impacting our ability to obtain light curves with a photometric precision near the ones obtained for the other stars. Furthermore, for Star 2 (7′′,-1) and Star 3 (8′′,-0.5) we can clearly see that the error bars are substantially smaller for the points obtained with the dynam star tracking method.

Following that, we also see that the circle mask is much better than the shape one, for Star 3 (8′′,-0.5), assuming dynam star tracking methods. We believe that this occurs due to the shape mask having an irregular shape and thus it’s easier for it to leave the image region, leading to a stricter constriction in its size, when compared with the circular mask.

Analysis of the optimal methods

The last characterization made with this data set will be through the methods that were able to produce the best light curves. As we could not find a single optimal combination for both the target and the background stars, we used the optimal for each of the cases, i.e. a combination of shape-dynam-offsets for the target star and a combination of circle-dynam-dynam for the background ones, as archi can use different methods, independently, for both the target and background stars. This comes from the fact that the target star remains centered and sampled in the same pixels, while the background stars rotate and travel throughout many different pixels. Furthermore, we also tested with a background grid with 600 points and without it, in an attempt to find artifacts introduced by its usage, whilst maintaining the same set of combinations.

In Fig. 6 we find that the background grid in use does not impact, significantly, the flux level, although it manages to reduce the measured noise. Interestingly, we can find a sinusoidal signal in the background light curves, most noticeable on Star 1 (5′′,+1). In Silva (2019), we found that this signal was induced by the irregular shape of each star’s Point Spread Function, which was crossed by the background stars during their rotation. In this image we cannot find the same signal for Star 2 (7′′,-1), due to it being brighter than the target star, and thus the PSF’s impact is not as prominent.

Furthermore, when this contamination is not as present we find that the CDPP shows values close to the ones exhibited on the central star, with similar confidence intervals.

4.2 Impact of the target star’s magnitude

Since the exposure time is set to be the ideal one for the central, target star, we can find cases in which the background stars are brighter than the target and thus, they might impact negatively the star detection algorithms. To see how this can impact our capability of detecting the background stars, we shall now look into Data Set B.

Starting with Data Set B1, in Fig. 7(b), the case in which the target star is the brightest star, the exposure time is shorter, thus not allowing to collect as much photons for the background stars which leads to us not being able to detect them. The sole exception is the faint star found above the central star, since it’s the brightest background star. If we now use a target star not as bright but maintain the
magnitudes of the background stars, as seen in Fig. 7(c), we are capable of detecting another star, although it’s barely visible.

Lastly, if we reduce even further the central star’s magnitude to 12 mag, in Fig. 7(d), the closest star to the target is still not detected. The long exposure time needed to observe the target star leads to the saturation of Star 3 (8″, -3) on top of the image. As the nearest star to the target, Star 1 (5″, -1), is equal to the one in Fig. 7(c), from data set A, we would also expect to detect it. However, that is not the case due to a limitation of the default image pre-processing normalization routine.

However, if we use an iterative normalization routine, as described in Section 3.1.2, we are able to detect the fainter stars, with the downside of an higher computational cost. Even though this cost is not extremely evident for smaller background grids, it can quickly ramp up for larger ones, as we are doubling the number of times that we must apply our analysis routine.

From Figure 8 we see that, with the iterative normalization routine, we are able to detect the previously undetected stars, seen in Figure 7. However, as we are removing regions of the image, there is always a chance that we will leave some residual pattern, that can be detected as a star. On top of that, the over-usage of this iteration process, may lead to cases where some regions of the background start being above the binary threshold in use. As the faintest stars are able to detected, is unlikely that stars not accounted for will contribute to signal found on the target star.

Attempts to work with the logarithm of the images, to avoid the double analysis of the images, lead to an increase of the rate of false positives. We found that, with different types of thresholds, even though we were able to detect the fainter stars, we got a greater number of false positives near the edges of the image. The application of filters to smooth the noise, coupled with an adaptive threshold might be able to reduce the false positives. Furthermore, as all tests have been performed with simulated data sets, we must still to validate the current routines over real data.
Figure 6. Light curves extracted, for Data Set A, without a background grid, in black, and with a grid of 600, in red. For the target star we used a combination of shape-dynam-offsets and, for the background ones a combination of circle-dynam-dynam.

Figure 7. Optimal masks for data set A and B. With Data Set B we can see the limitations of the basic normalization routine, as the faintest stars are not found, due to the presence of brighter stars in the image.
4.3 Impact of the star’s rotation

As we have mentioned within this paper, we expect to find larger errors in the light curves for the orbiting stars, due to the continuous change in the pixels that are collecting light from the star. A portion of this degradation might come from the Flat Field correction applied to the images. Although expected to be a lower noise source, the FF correction being applied to the target stars, where it is assumed a single spectrum type, will not match the background star.

In order to quantify the loss in precision for those stars, when compared with the target star, we shall use Data Set C, which has 3 stars all with the same magnitude, as described in Appendix A. We had to decrease the magnitude of the central star PSF that is, as previously discussed, crossed by the masks of the background stars. Some fluctuations can also be found for Star 3, but not as clearly. The light curve from Star 3 (8′′,0) also presents slightly lower flux values, although that was expected due to the limitations imposed to the mask’s size by the closeness to image’s border. It’s also noteworthy that, once again, we find that the shape mask performs much worse than the circle mask, for this star.

If we now compare the photometric precision from the target star to the background ones, in Table 2, we see that the former achieves a photometric precision 2 to 3 times better than the ones obtained for the background stars. The errors in the photometric precision also take smaller values for the light curve from the target star.

Within the background stars we find similar noise values among the stars, although Star 3 (8′′,0), with a circular mask, has lower error bars in the photometric precision. Furthermore, we can also see that the error bars suggest that the target star is less affected by the noise sources, which we can also verify with a visual comparison of the light curves.

Lastly, it’s important to notice that even though the chosen combination of methods is not the optimal one for the target star, the photometric precision in it surpasses the ones obtained for the background ones.

4.4 Contamination of the target by background stars

In order to understand if the background stars will induce spurious signals on the target’s star light curve we made use of Data Set D1 and D2, shown in Appendix A. The former, has a planet orbiting the background star, Star 1 (5′′,-2), whilst the latter does not have it. If we subtract the light curve extracted from the target star of D2 from the one extracted on D1, we can see the portion of the astrophysical signal that is in fact introduced by nearby transits. If no contamination exists, then one would expect to find that both light curves are equal, without exhibiting any differences between them.

From Fig. 10 one can see that the residual signal is not zero but instead shows a behavior concordant with the transiting events from the background star. Furthermore, the fact that we can see effect of the background star’s transit dip also serves as an indicator of its impact, even when there is no transit occurring.

With this, we can conclude that it is possible to find...
Figure 9. Light curves extracted, for Data Set C, without a background grid, a dynam initial detection method and a dynam star tracking method. In Table 2 we can find the photometric precision and corresponding value of the error bars in its value.

Figure 10. Comparison of the signal found on the target star, with the transits of the background one. In black we have the normalized light curve from the background star, on Data Set D1, where we can clearly see a transiting event. In the red dashed line, we have the absolute difference between the normalized light curve from the central star, when there is a transit on the background star (D1) and when there isn’t one (D2).

spurious signals on the target’s light curve, induced by other nearby stars. As archi gives one the possibility of extracting those light curves, it allows for a manual analysis to validate if the astrophysical signal in question might derive or not from contamination of the background stars. At a later stage, those light curves could also be used to correct the contamination that each star applies on its neighbors, similarly to what was done for the K2 mission (Luger et al. 2018).

5 CONCLUSIONS

Within this work we have presented archi, an open-source python pipeline that is capable of extracting light curves from the background stars of the CHEOPS mission. This new pipeline is capable of detecting and properly tracking the rotating stars, extracting light curves with a photometric precision 2 to 3 times worse than the one obtained for the target star. By resampling the original image we managed to create a virtual reduction in pixel size and thus achieved a reduction of the noise metric for the central star. Despite the good results for that star, the same was not verified for the background stars, for whom this grid would yield almost no gains.

Through an empirical analysis we found that the benefits for the target star are also capped, i.e. using grids with a side with more than 1800 points does not reduce more the noise but, instead, the noise oscillates around an “equilibrium” value. This equilibrium stage is met for the smallest possible grid, with 600 points, which was also deemed the grid’s size that minimized the computational cost whilst providing light curves with that minimized the noise metric (CDPP).

We have also tested archi in edge cases, i.e. saturated stars or very faint stars. Under cases where we find a saturated star, due to one of the background stars being brighter than the target, we can still detect fainter stars, albeit with an higher computational cost for the pipeline. We believe that the normalization routine, that represents the larger bottleneck, especially with larger background grids, can be improved by filtering the images to reduce noise and con-
sequently, the number of false positives. Furthermore, if we replace the binary threshold with an adaptive one coupled with smoothing filters, could be able to give better estimates of the masks.

We have also found that in some cases the background stars could contaminate the target’s light curve, imprinting variations that coincide with the transiting events of the background star. Since our pipeline is capable of extracting the light curves of background stars it allows to search them for possible causes of astrophysical signals. We postulate that a later stage it could be possible to use those light curves to remove noise from the target star, similarly to what was done for the K2 mission.

Lastly, when comparing the light curves from the target star against the official Data Reduction Pipeline, which is still under development, we found that archi is capable of outputting light curves with an equal or lower noise metric than the DRP, with a smaller 68% confidence interval.

ACKNOWLEDGEMENTS

We thank Mathias Beck, on behalf of CHEOPS SOC and David Futyan, for the work on the CHEOPSim tool. We also thank the anonymous reviewer for the comments that allowed us to improve the manuscript.

This work was supported by FCT - Fundação para a Ciência e a Tecnologia through national funds and by FEDER through COMPETE2020 - Programa Operacional Competitividade e Internacionalização by these grants: UID/FIS/04434/2019; UIDB/04434/2020; UIDP/04434/2020; PTDC/FIS-AST/32113/2017 & POCI-01-0145-FEDER-032113; PTDC/FIS-AST/28953/2017 & POCI-01-0145-FEDER-028953. S.S. acknowledge support from FCT through Investigador FCT contract n° IF/00028/2014/CP1215/CT0002 and in the form of an exploratory project with the same reference. O.D.S.D acknowledge support from FCT through national funds in the form of a work contract with the reference DL 57/2016/CP1364/CT0004.

REFERENCES

Abdul-Masih M., et al., 2016, The Astronomical Journal, 151, 101 Broeg C., et al., 2013, EPJ Web of Conferences, 47, 03005 Christiansen J. L., et al., 2012, Publications of the Astronomical Society of the Pacific, 124, 1279 Deleuil, M. et al., 2018, A&A, 619, A97 Fortier A., Beck T., Benz W., Broeg C., Cessau V., Ehrenreich D., Thomas N., 2014, in Ochsmann J. M., Clampin M., Fazio G. G., MacEwen H. A., eds, Space Telescopes and Instrumentation 2014: Optical, Infrared, and Millimeter Wave. SPIE, doi:10.1117/12.2056687 Futyan D., et al., 2020, arXiv:2001.05587 [astro-ph] Hoyer S., Guterman P., Demangeon O., Sousa S. G., Deleuil M., Meunier J. C., Benz W., 2019, arXiv:1909.08363 [astro-ph] Jenkins J. M., et al., 2010, in Radziwill N. M., Bridger A., eds, Software and Cyberinfrastructure for Astronomy. SPIE, doi:10.1117/12.856764 Koch D. G., et al., 2010, The Astrophysical Journal, 713, L79 Lugrè R., Agol E., Kruse E., Barnes R., Becker A., Foreman-Mackey D., Deming D., 2016, The Astronomical Journal, 152, 100

Luger R., Kruse E., Foreman-Mackey D., Agol E., Saunders N., 2018, The Astronomical Journal, 156, 99

OpenCV 2019, OpenCv python official python tutorial - Contour Features, Online, https://docs.opencv.org/3.1.0/dd/d49/tutorial_py_contour_features.html

Rando N., et al., 2018, in MacEwen H. A., Lystrup M., Fazio G. G., Batalha N., Tong E. C., Siegler N., eds, Space Telescopes and Instrumentation 2018: Optical, Infrared, and Millimeter Wave. SPIE, doi:10.1117/12.2311300

Rocha L., Velho L., Carvalho P., 2002, in Proceedings. XV Brazilian Symposium on Computer Graphics and Image Processing. IEEE Comput. Soc, doi:10.1109/sibgra.2002.1167130

Silva A. M., 2019, Master’s thesis, Faculdade de Ciências da Universidade do Porto, https://hdl.handle.net/10216/124209

APPENDIX A: DATA SETS IN USE

For this work we used 7 different data sets, for three distinct use cases. The first, Data Set A, was used to compare archi’s methods amongst themselves and with the DRP. The following three, Data Set B1, B2 and B3, were used to see how the pipeline behaves for different exposure times. Data Set C was created to study the difference between the precision obtained for the target star and for the background ones, when the magnitude of all visible stars is equal. Finally, Data Set D1 and D2 were used to show that background stars could contaminate the light curve from the target star.

We maintained the separation between the background stars constant, as seen in Fig. A1, albeit with different magnitudes, as one can see in Table A1, depending on the use case of the respective Data Set. Data Set B1, B2 and B3 all have equal magnitudes of the background stars, with the only difference being on the magnitude of the target star and, consequently, the exposure time of the images. In Data Set C we made the magnitude of Star 1 such that it was too faint to be visible in the images. Lastly, for Data Set D1 and D2 we have also made Star 2 and Star 3 fainter, so that they would not appear on the images and, in Data Set D1 we introduced a transiting event on Star 1.

Figure A1. Initial position of the stars, for all Data Sets.
Table A1. Mapping between the stars, seen in Fig. A1, their magnitudes in each data set and the separation from the target star, in arcseconds. It’s important to keep in mind that Data Set C has Star 1 set in such a way that it does not appear in the images. It’s important to keep in mind that, in this Data Set, Star 1 will not be visible in the images and, consequently, not detected by archi.

| Data Set | Star 0/Target [mag] | Star 1 [mag] | Star 2 [mag] | Star 3 [mag] | Exposure time [s] |
|----------|---------------------|--------------|--------------|--------------|------------------|
| A        | 12                  | 13           | 11           | 11.5         | 60               |
| B1       | 7                   | 13           | 11           | 9            | 22               |
| B2       | 9                   | 13           | 11           | 9            | 36               |
| B3       | 12                  | 13           | 11           | 9            | 60               |
| C        | 9                   | 15           | 9            | 9            | 36               |
| D1/D2    | 9                   | 7            | –            | –            | 36               |
| Distance to centre [arcsec] | 0   | 5.004        | 7.300        | 8.155        | –                |
| Transiting events | No   | In D1       | No           | No           | –                |

This paper has been typeset from a TeX/LaTeX file prepared by the author.

Figure A2. Stars present on Data Set C.

Figure A3. Stars present on Data Set D.