K2$^2$ — A PHOTOMETRY PIPELINE FOR THE K2 MISSION

**Mikkel N. Lund$^{1,2,*}$, Rasmus Handberg$^{1,2}$, Guy R. Davies$^{2,3}$, William J. Chaplin$^{2,3}$, and Caitlin D. Jones$^{2,3}$**

1Stellar Astrophysics Centre (SAC), Department of Physics and Astronomy, Aarhus University, Ny Munkegade 120, DK-8000 Aarhus C, Denmark; *mikkelnl@phys.au.dk and
2School of Physics and Astronomy, University of Birmingham, Edgbaston, Birmingham, B15 2TT, UK

Abstract

With the loss of a second reaction wheel, resulting in the inability to point continuously and stably at the same field of view, the NASA Kepler satellite recently entered a new mode of observation known as the K2 mission. The data from this redesigned mission present a specific challenge; the targets systematically drift in position on a ~6 hour time scale, inducing a significant instrumental signal in the photometric time series — this greatly impacts the ability to detect planetary signals and perform asteroseismic analysis. Here we detail our version of a reduction pipeline for K2 target pixel data, which automatically: defines masks for all targets in a given frame; extracts the target’s flux- and position time series; corrects the time series based on the apparent movement on the CCD (either in 1D or 2D) combined with the correction of instrumental and/or planetary signals via the KASOC filter (Handberg & Lund 2014), thus rendering the time series ready for asteroseismic analysis; computes power spectra for all targets, and identifies potential contaminations between targets. From a test of our pipeline on a sample of targets from the K2 campaign 0, the recovery of data for multiple targets increases the amount of potential light curves by a factor of $\geq$10.

Our pipeline could be applied to the upcoming TESS (Ricker et al. 2014) and PLATO 2.0 (Rauer et al. 2013) missions.

Keywords: asteroseismology — methods: data analysis — techniques: photometric — techniques: image processing — stars: solar-type

1. INTRODUCTION

K2 (Howell et al. 2014) is the continuation of the nominal NASA Kepler mission (Borucki et al. 2010; Gilliland et al. 2010b) which ended with the loss of a second reaction wheel in May 2013. The stability solution for the Kepler satellite is to balance in an unstable equilibrium against the Solar photon pressure and correct rolls with thruster firings, while pitch and yaw is controlled by the two remaining reaction wheels; this strategy allows for observations in fields along the ecliptic plane, with an observing length per field of close to 80 days. This time span is known as a “Campaign” (C), and is the analogue to the 3 month “Quarters” (Q) used in the nominal Kepler mission. In the nominal mission targets were designated using a Kepler Input Catalogue (KIC) number, which has now been replaced by the Ecliptic Plane Input Catalogue (EPIC) number.

The systematic pointing drift in the K2 observations, from the adopted stabilisation of the spacecraft, calls for new light curve correction methods. One such has recently been proposed by Vanderburg & Johnson (2014), and use the positions on the CCD as a function of time to decorrelate the induced variations in the light curve. The larger fields around targets in K2 — needed to account for the apparent movement of the target on the CCD — and the increased crowding from pointing toward the ecliptic means that often many stars are found in a given frame. This, combined with the potential lack of aperture masks from the Kepler team, necessitates the development of new methods to extract the flux and position of targets from custom apertures, and this in an efficient and robust manner.

The paper is structured as follows: In Section 2 we describe the steps taken in our light curve construction, starting from raw Target Pixel Files (TPF) and going to the definition of pixel masks and extraction of target positions and light curves. Section 3 pertains to the correction of the light curves from the time dependent movement on the CCD; here we describe both our version of the 1D self flat-fielding introduced by Vanderburg & Johnson (2014) in Section 3.1, and our suggestion for a 2D approach in Section 3.2. In Section 4 we present results from a test of our pipeline on a target sample during C0, and conclude in Section 5.

2. LIGHT CURVE CONSTRUCTION

The nominal Kepler mission delivered a pixel aperture (a mask) where the chosen pixels optimised the mean signal-to-noise ratio (S/N) based on estimates of the pixel response function (PRF) and information from the KIC (Bryson et al. 2010; Jenkins et al. 2010). This mask could be used to construct custom masks by adding or removing pixels to the starting mask based, for example, on the amount of flux in the pixels. This procedure was adopted in the KASOC filter pipeline (Handberg & Lund 2014) using the routine developed by Mathur et al. (in preparation). Masks are no longer delivered, at least not for the data releases made to date, which calls for a new method to define pixel masks. Masks constructed from ranking pixels in order of their S/N, and then including the number of pixels which optimises, for instance, the combined differential photometric precision (CDPP) noise metric (Gilliland et al. 2011; Christiansen et al. 2012) or the mean S/N could run into problems if signals from other stars are not removed; this is especially difficult if there are secondary objects in close proximity to the primary target.

In the following we describe our pipeline for the con-
We go through the different steps in the K2P pipeline below. In all examples times will be in truncated Barycentric Julian Date (TBJD)\(^1\) given as BJD − 2400000.

### 2.1. Background estimation

As the initial step of K2P\(^2\) we estimate the sky background as a function of time, because this contribution is unaccounted for in the flux from the raw K2 target pixel data. For each time step we calculate the mode of the flux kernel density estimation (using Scott’s (Scott 1979) rule for setting the bin width) from all pixels as a uniform background flux across a given image. The sky background level is far from constant, but increases gradually (by around ~25\% in C0) over the course of a campaign; a typical example can be seen in Figure 1. In C0 the background level was further increased for many channels by the antipodal ghost image of Jupiter as it fell on one of Kepler’s dead modules\(^3\). The change in background levels can largely be attributed to changing levels of stray light entering the photometer from the change in angle between the Sun and the photometer, and is thus additive. Secondary changes might come from changes in focus as the heating of the spacecraft varies.

If the background level variation is unaccounted for it will appear in the extracted light curve; it is preferential to isolate this component and separate it from trends caused by the degraded attitude.

### 2.2. Summed image

For setting pixel masks we create a summed image. Here, frames are co-added after first having subtracted the corresponding sky background levels (see Section 2.1). We make use of the quality flags available in the pixel data fits files (Fraquelli & Thompson 2012), and ignore all frames with a flag indicating any non-optimal data. The effect of neglecting this is illustrated in Figure 2. Including frames with bad quality flags, for instance when reaction wheel momentum dumps are made, results in the creation of a shifted ghost image. If a summed image including a shifted ghost image would be used in setting masks, these would be much larger than needed and would essentially only add noise for the majority of the time series. It would also be difficult for an automatic routine, that can separate close targets, to identify the ghost image as belonging to the main target rather than being a target in its own.

### 2.3. Pixel mask selection

To fix the masks we first select which pixels can be included in a mask by setting a flux threshold. The threshold is obtained as the median absolute deviation (MAD) of the summed image flux distribution which falls to the left-hand-side of the mode of the distribution. Only the left-hand-side of the distribution is used as the right-hand side is influenced more strongly by the stellar flux.

On the pixels with flux levels above the threshold we run an unsupervised clustering algorithm to locate targets in the frame and set individual masks for these. Specifically, we use the *density-based spatial clustering of applications with noise* routine (DBSCAN; Ester et al. 1996) as implemented in the Python-based library Scikit-learn\(^3\) (see Pedregosa et al. 2011). DBSCAN only takes two input parameters: a neighbour radius \(r_0\), and a minimum number of points needed to form a cluster \(N_{\text{min}}\). Given the regularity of the pixel grid,

---

\(^1\) http://keplerscience.arc.nasa.gov/K2/C0drn.shtml

\(^2\) this differs from the Barycentric Kepler Julian Data (BKJD) given as BJD − 2454833.

\(^3\) http://scikit-learn.org/

---

![Figure 1](image1.png)

*Figure 1.*: Kernel distribution of flux within the pixel frame as a function of time during C0 (here for EPIC 202062417). The color scale goes from light for a low flux level to dark for a high flux level; the red line indicates the distribution mode. Times with quality flags indicating contamination of any sort have been excluded (Fraquelli & Thompson 2012). The presence of a ghost image of Jupiter elevates the background flux between 56728 – 56788 TBJD, with high flux-spikes at the beginning and end of this interval where Jupiter enters and exits the focal plane and making specular reflections.

![Figure 2](image2.png)

*Figure 2.*: Summed image from short-cadence C0 data of EPIC 202062417, with the background mode (see Figure 1) subtracted from individual frames. Colour scale is on a logarithmic scale going from light (low flux) to dark blue (high flux), and negative levels are truncated to 0. Left: Summed image with frames having a bad quality flag removed. Right: Summed image with all frames included; here are ghost image of the brightest targets appear shifted approximately two pixels up and five pixels to the right.
these parameters can be set optimally a priori to yield a desired output. An advantage of the DBSCAN routine is that it does not need a predefined number of clusters, and that the clusters can have very irregular shapes — allowing it to encompass the spatial distribution of flux from a star on the CCD in K2, which depends both on time and position on the focal plane.

The working principle of the DBSCAN is, briefly: (1) select at random a point, with “points” being the pixels with flux above the threshold; (2) check how many other points \( N_c \) are within the neighbourhood radius \( r_c \) of the selected point; (3) if \( N_c \geq N_{\text{min}} \) the point is designated as a core point and the start of a cluster, otherwise, if \( N_c < N_{\text{min}} \), it is (at this step) designated as a noise point; (4) step (2) is now run on points within \( r_c \) of the first point, and so on for their respective neighbourhood points, and points are added to the first cluster until no more points are density reachable — that is, can be connected by a chain of points to the initial point seeding the cluster; (5) a point that falls within \( r_c \) of a cluster core point, but which has \( N_c < N_{\text{min}} \) in its own neighbourhood, is designated as an edge point to the cluster. Note that if such an edge point was the first considered by the routine, it would have been flagged as a noise point, but it will change status later in the routine if found within \( r_c \) of a cluster core point; (6) when no more points can be added to the first cluster, one of the remaining points is selected at random and the steps are run through anew. This continues until all points have a designation.

An illustration is provided in Figure 3 where we set \( r_c = \sqrt{2} \) pixels and \( N_c = 3 \). Each of the clusters returned are seen as a target, with the core and edge members of the individual clusters defining the outer boundary of the masks of the targets. Edge members within reach of more than one cluster could belong to either one of the clusters, and the membership of such a point would be determined entirely by the random initialisation of the routine. Core members, on the other hand, can be assigned clusters with full determinism, and will always group in the same way. We find, however, that the gain from a larger mask, which includes both core and edge members, outweighs the potential ambiguity and loss of repeatability from including a point that could belong to more than one cluster. In order to make the clustering reproducible we chose a fixed random seed for the algorithm\(^4\), which ensures that the clustering and designation of point will stay the same for a rerun with the same settings.

2.4. Saturated targets

The setting of masks for saturated targets calls for some extra attention. The saturation limit is at a Kepler magnitude\(^5\) of \( \text{Kp} \sim 11.3 \) (Gilliland et al. 2010a), and saturated targets will typically have pixel column trails along which flux spills, or bleeds. If the ends of these trails fall outside the mask the variability in the flux will be missed, resulting in a high-flux truncation of the light curve. The bleed-out is position dependent from the varying pixel sensitivities across the focal plane, but in K2 it will also depend on time, because the targets now have a time and position dependent movement on the detector. This results in a even poorer predictability of the amount of bleed-out.

An optimum inclusion of bleed-out trails is particularly difficult if the trail extends to other targets, or reaches the detector edge; in such cases a trade-off must be made between the amount of flux that can be included from the main target and the contamination from neighbouring targets.

We have implemented the following procedure for dealing with saturated targets (see Figure 6): For a given target we compute for each pixel-column, using pixels in the target’s mask, the ratio between the absolute value of the median of the first differences in the flux counts of the pixels and the maximum flux count of the pixels. A low value of this ratio indicates a small relative variability in the flux counts, as would be the case for a near-constant flux level in a column with many saturated pixels. If the ratio is below 1\%, and the median of the pixel flux counts (still only for the pixels in the mask) is equal to or larger than half of the maximum flux count for the entire mask, the column is taken as having saturated pixels. The restriction on the median of the flux counts ensures that columns containing many pixels with flux levels close to the background, where the relative variability also is small, are recognised as non-saturated. For the columns identified as saturated we then add pixels to the mask if these have counts above the flux threshold used in Section 2.3. This could potentially result in pixels belonging to both a saturated target as well as a nearby secondary target.

For the brightest and most saturated targets, with bleed-outs spanning many tens of pixels (e.g., EPIC 202061312) and with multiple secondary targets in the near vicinity the mask should be defined manually — as was done, for instance, for the 16 Cyg stars in the nominal Kepler mission (Lund et al. 2014; Davies et al. 2015).

2.5. Separating close targets

After a set of clusters has been identified there is still the possibility that a given cluster might encompass two or more stars if these lie close to each other. To separate such targets in a given cluster we run an algorithm often used in image segmentation problems known as the watershed method (Beucher & Lantuejoul 1979; Beucher & Meyer 1993), as implemented in Scikit-image\(^6\) (see Van Der Walt et al. 2014).

\(^4\) specifically we used the seed 1138 (see Lucas 1977).

\(^5\) nearly equivalent to an R band magnitude (Koch et al. 2010).

\(^6\) http://scikit-image.org/
The idea in a watershed algorithm is to find the line(s) between two or more regions, that may be seen as topographical surfaces; considering two neighbouring catchment basins that are flooded with water, the watershed will be the line where water levels meet.

To transform the pixel clusters to a topographical relief each point in a given cluster is assigned a value from the metric given either by the negative of the euclidean distance to the nearest background point (i.e., a point not in the specific cluster) or the negative value of its flux. This results in cluster points close to the edge having low negative values while central points of the cluster, which are further away from the background and generally have higher flux levels, have high negative values; this constitutes the catchment basins. If a cluster includes two or more stars that are not completely covered by a common envelope they will have distinctive central dips in both the distance and the flux metric. If the stars share a common envelope (seen if the stars are very close, or if one star greatly outshines the other) the flux metric is superior in making distinctive dips for the two (or more) stars; the distance metric will rather make a central dip for the whole region covered by the common envelope. As the default we use the flux metric to separate targets.

In the adopted watershed algorithm we first identify the local minima of the metric used and then use these as markers for the centres of the catchment basins which are then flooded to find the watershed lines. To avoid noise peaks being considered as markers we first smooth the surface with a 2D Gaussian filter, and then locate the most prominent minima — these are then fed as markers to the watershed routine. We now have pixel masks for all targets in a given frame.

Following the method outlined in Sections 2.3-2.5 for setting the pixel mask, we obtain for a sample of 4691 targets observed during C0 (see Section 4) mask sizes as a function of magnitude (see Section 2.6) as given in Figure 7. Here we note a slight gradient in the mask size as a function of an-
21
21
20
12
11
18
16
41
31
16
41
30x184
Mask size (pixels)
150000
100
120
60
80
20
46x202
gular distance to the space craft bore sight for a given \( \tilde{K}_p \) gives the mask sizes from Aigrain et al. (2014). Markers filled with black gives the pixels included in the mask after check for saturation bleed-out; in the given case the whole of column 13 is added to the mask of the primary target.

Figure 6.: Addition of pixels in bleed-out columns for the saturated target EPIC 202062417, using short-cadence C0 data. Different lines correspond to different X-pixel columns, see insert. Red circle (green square) markers indicate pixels in (not in) the mask of the main target (see Figure 2). Markers filled with black gives the pixels included in the mask after check for saturation bleed-out; in the given case the whole of column 13 is added to the mask of the primary target.

Figure 7.: Mask size as a function of the proxy \( K_p \) (see Equation 2.2) for a sample of 4691 C0 targets (see Section 5). Black markers indicate the median \( K_p \) for each of the discrete mask sizes; the colour coding indicate the angular distance for each target to the space craft bore sight; the red dashed line gives the mask sizes from Aigrain et al. (2014).

gular distance to the space craft bore sight for a given \( \tilde{K}_p \); this is as expected because the arc traced in apparent movement on the CCD from the roll of the spacecraft increases linearly with distance from the bore sight. The scatter in this relation will have contributions from the dependence of the degree of flux-smearing on the target position on the focal plane, and the uncertainty of the determined magnitude. For comparison we also show the magnitude dependence of aperture sizes from Aigrain et al. (2014), where the authors use circular apertures/masks.

2.6. Target magnitudes

Our pipeline enables the extraction of data for multiple targets in a given frame, but from the information in the target pixel data we only have a \( K_p \) magnitude, \( K_p \), for the primary target. First, however, it should be noted that when targets were proposed for C0 the EPIC did not exist. Therefore, a magnitudes given in the EPIC\(^7\) for a given C0 target is the one provided by the principal investigator proposing the target, rather than one computed by the \( K_p \) team. For the same reason no information is given in the \( K_p \) entry of the EPIC for C0, which is suppose to contain information on the data used to compute \( K_p \)— one should therefore consult the proposal of a given target to assess how the magnitude was constructed. For the sample of targets we have analysed, viz., the proposal GO1038 (see Section 4), it turns out that the EPIC \( K_p \) magnitudes are given by \( J \)-band magnitudes from the Two Micron All Sky Survey (2MASS; Skrutskie et al. 2006). To transform these \( J \)-band magnitudes to more proper \( K_p \) magnitudes we use the transformation from Howell et al. (2012) between \( K_p \) and 2MASS \( J - K_s \) colors.

In order to investigate how parameters such as mask sizes and noise measures vary with magnitude we need a way to estimate \( K_p \) for all targets in a given frame. We approximate \( K_p \) by the proxy \( K_p \)-magnitude \( \tilde{K}_p \) defined as:

\[
\tilde{K}_p \equiv 25.3 - 2.5 \log_{10}(S),
\]

where “\( S \)” denotes the median of the flux time series extracted for the target (in units of \( \text{e}^{-}/\text{s} \)). The correspondence between \( K_p \) and \( \tilde{K}_p \) is shown in the left panel of Figure 8; some of the scatter in this relation will originate from the scatter in the mask size versus magnitude relation (see Figure 7), and a variation in pixel sensitivities between targets. We note that Aigrain et al. (2014) defines a proxy \( K_p \) magnitude in the same manner and also find an offset of \( \sim 25.3 \).

As a means of identifying targets falling within a given mask (see Section 2.7) we use the USNO-B1.0 catalogue (Monet et al. 2003), which is an all-sky catalogue with a completeness down to \( V = 21 \). We would like a measure of \( K_p \) for all targets from the USNO-B1.0 catalogue within a given frame, because this is used in the identification of targets (see Section 2.7). In addition we can estimate potential contaminations when multiple targets fall within the same mask. For each of the identified targets from the USNO-B1.0 catalogue that fall within a given mask, we first compute the magnitude \( R_B \) from the USNO-B1.0 \( R \)- and \( B \)-band magnitudes:

\[
R_B = \begin{cases} 0.1B_{\text{mag}} + 0.9R_{\text{mag}}, & (B_{\text{mag}} - R_{\text{mag}}) \leq 0.8 \\ 0.2B_{\text{mag}} + 0.8R_{\text{mag}}, & (B_{\text{mag}} - R_{\text{mag}}) > 0.8 \end{cases}
\]

(2.2)

According to Brown et al. (2011) this corresponds to the way \( K_p \) magnitudes, \( K_p \), are calculated in the KIC if only \( R \)- and \( B \)-band magnitude are available. We define the following relation as a second proxy \( K_p \)-magnitude:

\[
\tilde{K}_p \equiv R_B - 0.33.
\]

(2.3)

The correspondence between \( K_p \) and \( \tilde{K}_p \) is shown in the right panel of Figure 8.

The relation giving the \( \tilde{K}_p \) proxy has the smallest amount of scatter, and will be used to relate mask sizes and noise measures to magnitude; \( \tilde{K}_p \) will be used in the identification of targets, and estimation of contaminations. An advantage of having both \( K_p \) and \( \tilde{K}_p \) is also that a large discrepancy between the two measures can be used to identify targets where the mask is either much too large or small.

The offsets for both \( K_p \) and \( \tilde{K}_p \) were estimated in a Bayesian manner using the affine invariant \textit{emcee} sampler

\[ \text{http://archive.stsci.edu/k2/epic.pdf} \]
(Foreman-Mackey et al. 2013), and given by the median of the marginalized posteriors; the uncertainties were obtained from the 68% highest probability density of the marginalized posteriors.

### 2.7. Locating main and secondary targets

In K2 a standardised mask is no longer delivered for the main target, at least not in the data releases so far. This, combined with the increased crowding in the equatorial pointing and larger frames, makes it more difficult to assert which target is the main target. Also, the primary target is sometimes fainter than secondary targets in the frame. A starting point for locating the primary target is the assumption that it is (approximately) centred in the frame, but still it will be difficult to use this exclusively in crowded fields. The target pixel files from K2 do deliver a world coordinate system (WCS; Greisen 2006) metric in the FITS format. The WCS from K2 data release 2 is fairly well calibrated (not available in the engineering data as shown in Figure 9). Here we have marked the positions of all targets from the USNO-B1.0 catalogue (Monet et al. 2003) from using the WCS to pixel coordinates; it is clear that the WCS delivers a reasonable transformation, generally within two pixels of maxima in the summed images. An advantage of our pipeline is that masks are defined for all targets in the field (unless they are too faint), and so identification can be made at a later stage.

So far we have made identifications using the Python module Astroquery (Ginsburg et al. 2013), together with the WCS module in the Kaptayn package. This enables us to link targets with objects from the USNO-B1.0 catalogue. The procedure for the identification of targets is as follows: (1) load sky coordinates for all targets located within a circular region which fully contains the frame for the EPIC target in question; (2) Transform the sky coordinates of these targets to pixel positions using the WCS from the K2 target pixel file; (3) Compute a proxy for the Kepler magnitude, $K_p$ (see Section 2.6 above); (4) Locate maxima in the summed image (Section 2.2) where a 0.5 pixels wide Gaussian smoothing has been applied; (5) Compute all $X$- and $Y$-pixel differences between the targets and the maxima of the summed image; (6) Run a DBSCAN clustering on all pixel differences within ±5 pixels in both the $X$- and $Y$-direction, with the clustering parameters set to $r_c = 0.5$ pixels and $N_c = N_{\text{max}}$. The value of $N_c$ will initially be the number of identified local maxima and will iteratively be decreased until a cluster is identified in the differences (a “difference cluster”); (7) If more than one difference cluster is found within ±5 pixels, we choose the cluster with the lowest mean $K_p$ magnitude; (8) As the correction that should be applied to the WCS transformation we take the weighted average of $X$- and $Y$-differences in the difference cluster, using one over the $K_p$ magnitudes as weights (see Figure 9). Note that this correction only includes translation, but ignores rotation. This could be amended by using a pattern matching algorithm (see, e.g., Spratling & Mortari 2009), but the offsets are low enough that this can be safely omitted; (9) The target from USNO-B1.0 with corrected pixel coordinates closest to the median centroids of identified target clusters is used to identify the cluster. Here we also note if other targets fall in the mask of a given target cluster.

### 2.8. Target flux and position

The above steps were concerned with the creation of pixel masks for all the different targets in a given frame. For all these targets we compute the position and flux as a function of time — this will be used later for the correction of the spacecraft roll. We use weights, $w_i$, on the pixels in the individual masks when extracting fluxes and calculating the target position via the centroid (CEN) with $(X,Y)$ components given as

$$
\text{CEN}_X = \frac{\sum_i p_i w_i X_i}{\sum_i p_i w_i} , \quad \text{CEN}_Y = \frac{\sum_i p_i w_i Y_i}{\sum_i p_i w_i} .
$$

(2.4)

Here $p_i$ denotes the flux for the $i$th pixel in a given mask, and $X_i$ and $Y_i$ denote the coordinates of the pixel. We have tried the following four different weightings: (1) all weights are set to $w_i = 1$, giving an “in/out” mask where all pixels influence the position and flux with equal weight; (2) weights are given as the exact euclidean distance between a pixel in the mask $(X_i, Y_i)$ and the closest background pixel $(X_{i,b}, Y_{i,b})$:

$$
w_i = \sqrt{(X_{i,b} - X_i)^2 + (Y_{i,b} - Y_i)^2} .
$$

(2.5)

Such a weighting reduces the sensitivity of the extracted positions to the exact mask configuration, where, for example, a high spatial frequency of the mask from pixels at the mask

---

Figure 8.: Relation between proxy Keplermagnitudes $K_p$ (Equation 2.1) and $K_p^2$ (Equation 2.3) and the nominal Kepler magnitude, $K_p$, computed from 2MASS $J - K_s$ colors. The dashed lines give the 1:1 relation. For 0.5 magnitude bins in $K_p$ the median $K_p^1$ and $K_p^2$ values are given by a red marker.
edges could result in an unwanted flickering in the extracted parameters for a uniformly weighted “in/out” mask. In many ways this resembles the weighting done naturally when using pixel response functions (PRF) to extract centroids (Bryson et al. 2010), but without the need to optimise for centroid and total flux using a parametrised function. The use of Kepler calibrated PRFs is further complicated by the fact that the pointing jitter (now with an attitude control bandwidth of 0.02 Hz until C3, where it will be increased to 0.05 Hz, which is half of the bandwidth of nominal Kepler observation) and systematic movements within a cadence are different in K2 from the nominal Kepler mission. Also, for saturated targets the parametrisation fails to represent the flux distribution, and the PRFs are only defined for long-cadence (LC) observations; (3) weights are set to create a soft edge on the mask, with a uniformly weighted central region. This is accomplished by dividing every pixel into $11 \times 11$ subpixels, each of which is assigned a weight given by Equation 2.5 and normalized by 11.3. If this normalized weight is above 1 it is set to 1. This results in a mask edge of just over 1 pixels width where the weight gradually increases from one over 11.3 to 1; (4) weights from (2) are used for the centroid calculation, and weights from (1) are used for the flux extraction.

We compared the different weighting schemes in the power spectrum, in the centroids, and in the final corrected time series. No noticeable difference was found between schemes (1), (3), or (4); with this in mind we opt for the simplest scheme, i.e., (1). We also note, first, that scheme (2) gives centroid values with a lower point-to-point scatter; this might be of use for future (from C3) 2D corrections of short-cadence (SC; $\Delta t \approx 1$ min) data, but seems to have little influence on the 1D correction. Secondly, the flux from scheme (2) retains the greatest signal from the spacecraft movement, which is evident in the power spectra from this method. This might be expected from the peaked flux weighting, which increases the sensitivity to the spacecraft movement — a flat weighting is preferable for the flux extraction. Considering the choice between scheme (1) and (3) for extracting the flux, both of which have a predominantly flat weighting, (3) reduces the risk of contamination between targets. Scheme (1), however, makes it easier to identify any contamination that might occur in any case. For simplicity we opt for scheme (1) for both the position and flux extraction, but consider using either scheme (4), or a combination of (3) and (1), for data from future campaigns.

In the final step of extracting the flux from the defined masks we subtract the background level given as the mode of the flux distribution (see Section 2.1), but now only including pixels that are unassigned to a target mask. If a target is close to the edge, the weighting scheme given by Equation 2.5 will put highest centroid weight towards the edge. However, as long as the flux variability from position correlates with the measured centroid, the data from such edge targets should still be usable. The same goes for centroids from saturated targets, as long as the extracted centroids correlate with the relative flux variation they can be used in the correction; the absolute position of the target is of little importance.

2.9. Contamination between targets

Given that most EPIC frames contain multiple targets we compute a few statistics to ascertain the level of contamination between these targets. As a first metric we compute a contamination value, $C$, as one minus the flux ratio of the primary and all targets in the mask:

$$C = 1 - \frac{F_{\text{primary}}}{F_{\text{total}}} = 10^{0.4(m_{\text{total}} - m_{\text{primary}})},$$  \hspace{1cm} (2.6)$$

where $m_{\text{primary}}$ is the $\text{KP}_2$ magnitude of the brightest target in the mask; the total apparent magnitude, $m_{\text{total}}$, of the mask is given as

$$m_{\text{total}} = -2.5 \log_{10} \left( \sum_i 10^{-0.4 m_i} \right),$$  \hspace{1cm} (2.7)$$

Figure 9. : Correction of the WCS pixel positions. Left: Circles indicate positions of the targets from the USNO-B1.0 catalogue falling in the frame of EPIC 202127012, where the transformation from sky to pixel coordinates was made using the WCS metric from the K2 pixel target files; target magnitudes $\text{KP}_2$ (see Equation 2.2) are indicated on the color scale and by the marker size. These are plotted on top of the summed image from C0 data, with a flux scale going from white (low flux) to dark blue (high flux) and rendered on a logarithmic scale. The arrows give the estimated correction to the WCS pixel positions; red crosses indicate the identified maxima in the summed image that are used to estimate the correction. The primary target, i.e., EPIC 202127012, is close to the center of the frame and emphasised with a white “+” at the corresponding target position. Right: Position differences between USNO-B1.0 targets and maxima in the summed image. Black crosses are differences tagged as noise in a DBSCAN run on the differences; red circles give the differences of the identified cluster. The final magnitude weighted average of the difference cluster is given by the black “+”.

$$\text{KP}_2$$
where \( i \) runs over the number of identified stars falling within the given mask. Secondly, for a given frame we compute a target correlation matrix. The lower left half of Figure 10 shows the correlation between the targets’ power spectra (of the cleaned time series, see Section 4); the top right half gives the minimum distance between pixels belonging to each target pair. This correlation matrix can be used to easily ascertain the contamination between targets, and thus when extra care should be exercised in assigning a given signal to a given star.

3. CORRECTING THE LIGHT CURVE

We have combined the correction part of our pipeline with the KASOC filter (Handberg & Lund 2014), meaning that the corrections based primarily on the target movement on the CCD are combined with corrections made for long and short term instrumental trends — and this in an iterative manner. The integration with the KASOC filter also includes the iterative use of phase curve corrections, which is particularly useful for separating the flux variations from the target movement on the CCD from those of stellar variability with a strict periodicity (for instance the eclipses of a planetary or binary system).

Below we describe the two possible correction methods in the pipeline. For both methods it generally holds true that when the amplitude of the underlying stellar signal dominates the variations, such as in many Classical pulsators, the correction of the instrumental signal is less effective.

3.1. 1D correction

Our 1D correction draws heavily on the method presented by Vanderburg & Johnson (2014) — which these authors called a self-flat-fielding correction — which in turn make some use of methods developed for correction of Spitzer data (Knutson et al. 2008; Ballard et al. 2010; Stevenson et al. 2012). These methods use the correlation between flux variation and position on the CCD (from pixel sensitivity differences across the CCD), to correct the time series from the systematic \(-6\) hour variability.

We break the time series into segments that are corrected individually. This segmentation was implemented because even though the movements on the CCD generally follow a well-defined pattern (which depend on position on the focal plane), there are slow uncorrected drifts as a function of time (see Figure 11 for an example of this in C0). Currently, the times where breaks are introduced are determined manually, and are kept constant for all targets in a given campaign; we provide flags for the times where breaks are introduced in the final output.

For each segment, we start by identifying and flagging times during which a rapid positional change occurs, as the times when the time derivative of the change in centroid positions, i.e., the velocity, falls outside the range of five times the standardised MAD \( \Delta Y \) around the median velocity; these data points are then excluded in the following corrections.

We then apply a principle component analysis (PCA) on the \( X \) and \( Y \) pixel positions of each data segment. Before applying the PCA, we select which of the \( X \) and \( Y \) pixel positions should be retained in the estimate of the correction; only positions with a nearest neighbour at a distance less than four times the standardised MAD of all nearest neighbour distances are retained in the analysis. This is needed as the PCA otherwise is very sensitive to outliers. The PCA transformation of the retained positions to the coordinate system given by the two first principal components, helps to ensure that the relationship between the transformed pixel positions \( X' \) and \( Y' \) can be described as a single-valued function, which is needed for the following steps in the correction. It is, however, not always clear if the first or the second principal component should be used as the regressor. If, for instance, the relationship between \( X \) and \( Y \) pixel positions could be described as \( Y = X^2 \) (which is already a single-valued relationship) and the range in \( Y \) values is larger than the range in \( X \) values, then the first principal component would lie along the ordinate and consequently a transformation making this the regressor, that is, \( Y \rightarrow X' \) and \( X \rightarrow Y' \), would result in the multi-valued relationship \( X' = \pm \sqrt{Y} \). We decide which of the principal components is the best regressor, by running a LOWESS (Cleveland 1981, 1979) filter on the transformed pixel positions, using in turn the two principal components as regressors and computing the summed squared difference

\[ \text{MAD} = \text{scale factor} \times \text{standard deviation of normally distributed random variable} \]
Relative Flux (ppm)

Figure 12: Illustration of 1D correction for star 2 (TYC 1329-1325-16) in EPIC 202127012 (see Figure 5). Top: Uncorrected time series in flux relative to the median. Middle: Time series corrected for long term, short term, and positional trends. Bottom: Time series corrected for long term, short term, positional trends, and the phase curve constructed from the middle panel (see Figure 13).

Figure 13: Phase curve for star 2 (TYC 1329-1325-16) in EPIC 202127012 (see Figure 5). Top: Phase curve of uncorrected time series with the flux relative to the median (top panel in Figure 12). Bottom: Phase curve (black) of time series corrected for long term, short term, and positional trends (middle panel in Figure 12). From the black points we form the red phase curve via a moving median smooth; this smoothed phase curve is used in the iterative correction performed by the KASOC filter to obtain the bottom panel of Figure 12 (Handberg & Lund 2014).

$(\chi^2)$ between the filtered and un-filtered data. The principal component with the lowest $\chi^2$ is used as the regressor.

In the transformed coordinates we compute a smoothed version of the $Y'$ vs. $X'$ positions by again applying a LOEWESS filter. We then calculate the curve length along this filtered relationship using finite differences as the derivative of the curve and cumulatively integrate for the curve length using the composite trapezoidal rule. The curve length $s$ serves as the new 1D representation of the 2D stellar position on the CCD.

The correction to the light curve is then found from a LOEWESS filtering of the relative flux as a function of curve length, thereby capturing the average positional dependency of the flux level. In the correction step we make sure to remove any long term trends in the light curve to obtain the relative flux, as such changes will correlate poorly with the movement on the CCD. Some of the long term variability could in principle be caused by the slow drift of the target on the CCD (Figure 11), but could just as well be a separate instrumental effect — for instance from focus changes caused by heating of the mirror. The background flux level could also enter in the long term variability if this is not corrected for properly during the light curve extraction. We make the correction iteratively with a better separation between long term and positional dependent variations as the outcome.

In Figure 12 we give an example of the 1D correction for the C0 observations of star 2 (TYC 1329-1325-16) in EPIC 202127012 (see Figure 5); here we further include in the KASOC filter a correction for the dominating periodic signal by iteratively correcting by the phase curve of this signal (see Figure 13). The input period for this correction was determined from the autocorrelation function of the time series.

3.2. 2D correction

In our second approach we make a 2D histogram of the measured $X$ and $Y$ centroids of the star. In each bin we compute the median of the relative flux of points falling in that bin; this will capture the positional variation in the relative flux in a robust manner. In the reconstruction of the flux variability in the time domain we use a rectangular bivariate linear spline to interpolate between the bin centres. The reason for going to 2D is that flux variations also occur in the direction perpendicular to the overall roll motion (see Figure 14 for an example). Such variations are unresolved in the 1D treatment, because the scatter in the relative flux versus curve length is reduced to a line; one would therefore suspect that the 1D treatment will leave residuals in the corrected light curve that could be accounted for in a 2D treatment.

The most difficult aspect of the 2D binning is the choice of bin size. If the bins are too small the reconstruction of the flux variation will be noisy; one is effectively overfitting. On the other hand, if the bins are too large the reconstructed variation will be a smoothed version of the underlying variation, and significant residuals may be left in the light curve. The sensitivity to the bin size is largest for long-cadence (LC; $\Delta t \approx 29.4$ minutes) observations due to the smaller number of data points, and consequently larger variance on the median. The method is thus best suited for SC observations where the exact bin size is less influential on the reconstructed instrumental variability.

Depending on the shape of the stellar movement in the $X$-$Y$ plane it can be advantageous to transform the movements to a (predominantly) horizontal variation before making the 2D histogram — this could, for instance, be achieved by dividing the centroid $Y$-components with a smoothed fit to the movement in order to reduce the span of the histogram, and thus the size, of the histogram.

So far, in our testing of this method on SC data we have not found it to be preferable to the 1D method. This is likely a result of the current value of the attitude control bandwidth of 0.02 Hz (50 s), which is very close to the SC integration time. Because of the allowed amount of movement within a SC integration this will lead to a larger smear and variance in the bin medians. We expect this to improve from C3 onwards when the bandwidth will be increased to 0.05 Hz.

4. PIPELINE TEST

As a test of the pipeline we analysed the pixel frames of the 452 LC targets in the C0 proposal GO0118$^{10}$ ("Galactic

$^{10}$ http://keplerscience.arc.nasa.gov/K2/docs/Campaigns/C0/GO0118_Stello.pdf
Archaeology on a grand scale”; PI: Stello, D.). We also analysed the known transiting system WASP-85 (see Brown et al. 2014), which was observed in SC during C1.

Because our pipeline enables the extraction of data from several targets in a given frame, we ended with a total of 4691 targets from the GO0118 proposal, and thus light curves to analyse — this corresponds to a gain in the amount of data by a factor of \(\sim 10.4\), and this even when adopting a limit on the minimum number of pixels in a mask of 8 before a target would be considered.

### 4.1. The power spectrum

After data were extracted using the K2P\(^2\) pipeline, they were corrected with the KASOC pipeline (Handberg & Lund 2014) using the 1D correction method described in Section 3.1, and a frequency power density spectrum was calculated. The 1D correction removes most of the signal from the spacecraft roll, but residual spikes still often appear at harmonics of \(\sim 47.2281\ \mu\text{Hz}\). These spikes are damaging to any automated search for power; to remedy this we tested the effect of “cleaning” the residual spikes using a prewhitening routine (see, e.g., Ponman 1981; Belmonte et al. 1991), which removes all significant power in a \(\pm 1\ \mu\text{Hz}\) window around the residual spikes. For every window, oversampled by a factor of ten, we iteratively remove the frequency with the highest power-to-background ratio (PBR: the background is calculated as the median of the power within the window multiplied by \(-1.42\), which is the conversion factor between the median and the mean for a \(\chi^2\)-distribution) if this ratio has a false-alarm detection probability less than 10% (Scargle 1982; Appourchaux 2004; Lund et al. 2012). Besides the signal from the spacecraft roll, we also see a signal at \(\sim 5.92\ \mu\text{Hz}\) (equivalent to \(-1.96\) days); we suspect this signal originates from the periodic momentum dumps of the reaction wheels through thruster firings, which happens every two days (Howell et al. 2014), and enters the power spectrum via the spectral window (see right panel in Figure 15). The left panel of Figure 15 shows the efficiency of the procedure for removing the residual instrumental peaks from the power spectrum. Instrumental signals can still be seen in the cleaned power spectrum, but now with amplitudes low enough to allow the detection of asteroseismic signals.

### 4.2. High frequency photometric variability

To detect stellar oscillations in the frequency power spectrum it is important that the white (shot) noise level does not dominate the signal — this is especially true for the detection of low amplitude stochastic solar-like oscillations. It is thus of interest to know the characteristic levels of the short time scale (high frequency) noise in K2 LC data as a function of Kepler magnitude, or in our analysis \(K\_P\). We note, however, that a measure of the high-frequency noise is not necessarily tantamount to a measure of the constant power spectral density white noise level. For each of the targets in the sample we computed a proxy for the instrumental variability using the median of the absolute point-to-point flux difference of the KASOC corrected and cleaned time series; this proxy was coined the median differential variability (MDV) by Basri et al. (2013). As detailed in Basri et al. (2013) the MDV will on short time scales (with point-to-point being the shortest) be most sensitive to high frequency noise; variability on time scales longer than the LC sampling of \(\sim 29.4\) minutes will on the other hand contribute very little to the MDV. To enable a comparison of the MDV for K2 with that of the nominal Kepler data, we compute the point-to-point MDV for the set of 6210 LC targets from the Kepler APOKASC (Pinsonneault et al. 2014) data release 1 sample. In the KASOC filtering we used the following filter settings: \(\tau\_\text{long} = 3\) days and \(\tau\_\text{short} = 0.25\) days (see Handberg & Lund (2014) for details on these settings); for the APOKASC targets we used \(\tau\_\text{long} = 30\) days, which is a too long time scale for the duration of the K2 light curves. Figure 16 shows the resulting MDV measures as a function of magnitude for both the K2 and nominal Kepler targets. Our results from the nominal Kepler data are in overall agreement with the results presented in Basri et al. (2013). We find that at \(K\_P \lesssim 10\) the ratio between the median MDV in K2 and nominal Kepler falls below \(\sim 2\), and increases to \(\sim 10\) at \(K\_P \sim 14\). For the K2 values we further see an indication of a slight gradient in the MDV with angular distance to the bore sight for a given magnitude, which might be expected from the larger systematic imprint on the light curve further away from the bore sight. Comparing our values to those from Agrain et al. (2014) (their Table 1, 3-pixel radius masks) we find, as evident from Figure 16, an excellent agreement. We also computed point-to-point MDVs for our target sample as corrected in Vanderburg (2014)\(^1\), and find that the median binned values generally agree within a factor of two. For these comparisons it should be noted that we are unaware of how the authors of the comparison studies calibrated the Kepler magnitudes extracted from the TPD, if at all.

We note that a comparison of MDVs can not be seen directly as a comparison of the quality of the light curves and the corrections applied, because a measure like the MDV will depend strongly on the choice of free parameters in the correction. In Vanderburg (2014) the C0 light curves were corrected individually in three segments; the values from the mask with the lowest 6-hour scatter were adopted, trying 20 masks of different sizes; the fit to the flux versus curve length was made with a finer binning than in Vanderburg & Johnson (2014) — all of these tweaks will conspire to give a lower point-to-point scatter.

---

\(^1\) https://www.cfa.harvard.edu/~avanderb/k2.html
Visible at integers of \( \sim \) \( \mu \)Hz targets from the nominal Kepler space craft bore sight; filled black markers give the estimates for APOKASC the K2 sample, with the colour scale indicating the angular distance to the power spectrum and spectral window both of these were interpolated onto a common frequency scale using a smoothing spline interpolation.

If we assume the MDV scales linearly with the shot noise (virgin MDV) is, according to Figure 16, only about enough to encompass the movement of the target on the CCD; this in turn considerably increases the risk of contamination from other nearby targets. The use of clustering of pixels from the summed image for defining the masks better approximates due to the large amplitudes of the oscillations.

Concerning the construction of pixel masks we note that many of the published studies of K2 data apply circular masks. But, the flux distribution for a target in K2 is generally far from circular and symmetric, especially if a summed image is used. If a circular mask is used it needs to be large enough to encompass the movement of the target on the CCD; this in turn considerably increases the risk of contamination from other nearby targets. The use of clustering of pixels from the summed image for defining the masks better approx-
imimates the actual flux distribution of the target. For later versions of the pipeline we will investigate in greater detail if any weighting of the pixel masks can lead to a reduction of the high-frequency noise, e.g., as measured via the point-to-point MDV. In relation to this we will also test further the potential impact of a high spatial frequency of the derived pixel masks. More effort needs to be invested in improving the correction of instrumental trends via the 2D method. When data from C3 becomes available, where the fine pointing of the spacecraft should be improved, we will revisit this method in more detail. This could also include an implementation of the procedure outlined in Kjeldsen et al. (2013a,b). We will also continue to try and improve the 1D correction, that in our tests still seems to leave artefacts at harmonics of ~47.2281 µHz.

A better removal of these artefacts is clearly needed if an automatic search of asteroseismic power is desired, and simply masking the peaks in the power spectrum will only have a limited impact if the effect of the spectral window is neglected. Our attempt at cleaning the instrumental peaks did improve the power spectrum, but still could not fully remove the instrumental peaks and the window function persisted, which might be expected from cleaning a highly non-sinusoidal signal. As part of the correction we will look into measures other than the centroids for the position of the stars on the CCD; this could include the construction of a mean relative movement on the CCD from combining the measures of all targets in a given pixel frame. Also of interest is whether the housekeeping data from the Kepler spacecraft can be incorporated for a better overall positional correction. We will attempt to improve the treatment of saturated targets, which are difficult to deal with via the DBSCAN clustering routine. Aspects that should be improved here are, for instance, a better separation of targets that fall within or close to high-flux pixels from a saturated target.

We note that our method could potentially be used for dense fields including stellar clusters, and could also be applied to super-stamps from K2 and the nominal Kepler mission, as well as the upcoming TESS (Ricker et al. 2014) and PLATO mission, as well as the upcoming TESS (Ricker et al. 2014) and PLATO...
Figure 20. Phase curves for a subset of targets exhibiting transit-like features, rendered over two full phases with the first primary transit feature centred at 0.5; the approximate period for each target is given in the label of the abscissa. For the purpose of this illustration periods were estimated by eye from the light curves. In each panel we have indicated what we find the most likely USNO-B1.0 number for the target, the EPIC number for the star in whose pixel frame the target was found. In neither of the cases show did the target correspond to the target associated with the EPIC number.

During the development of K2P2 we tested the application of the pixel clustering on every time step for the pixel frame of a given target rather than using the summed image. A complication of this method over using the summed image is that the number of targets identified in the pixel frame varies slightly with time due to noise, and the cluster number of a given targets will also vary in time. From tests of this version of the pipeline on K2 engineering data, we found that using the pixel clustering on every time step could enable the detection of asteroids and/or comets (or other unidentified objects) as they passed through the pixel frame (see Szabó et al. 2015, for an analysis of asteroids found during the K2 engineering run). When scatter plotting centroid estimates for all identified targets (at a given time step) against time, moving targets such as asteroids make clear centroid trails that deviate from the horizontal trails of quasi-stationary targets such as stars. Identification and analysis of such centroid trails could lead to the detection and tracking of hitherto unknown asteroids/comets.

ACKNOWLEDGMENTS

We would like to thank all active participants at the first K2 data analysis workshop in Aarhus (Denmark, 2014) for many useful discussions on approaches to K2 data extraction and correction. A special thanks to Bram Buysschaert for giving us the idea of cleaning residual peaks from the power spectrum, to Daniel Huber for providing useful input on the EPIC, and to Hans Kjeldsen for commenting on the paper.

Funding for the Stellar Astrophysics Centre (SAC) is provided by The Danish National Research Foundation (Grant agreement no.: DNRF106). The research is supported by the ASTERISK project (ASTERoseismic Investigations with SONG and Kepler) funded by the European Research Council (Grant agreement no.: 267864).

W.J.C., G.R.D., and C.D.J. acknowledges the support of the UK Science and Technology Facilities Council (STFC).

This research took advantage of the SIMBAD and VizieR databases at the CDS, Strasbourg (France); NASA’s Astrophysics Data System Bibliographic Services (adswww.harvard.edu); arxiv.org, maintained and operated by the Cornell University Library; the USNOFS Image and Catalogue Archive operated by the

---

12 TESS: Transiting Exoplanet Survey Satellite; PLATO: PLAnetary Transits and Oscillation of stars.
REFERENCES


Cleveland, W. S. 1979, Journal of the American Statistical Association, 74, 829

—. 1981, The American Statistician, 35, 54


Fraquelli, D., & Thompson, S. E. 2012, Kepler Archive Manual, (KDMC-10008-004)


Scott, D. W. 1979, Biometrika, 66, 605


Sprenging, B. B., & Mortari, D. 2009, Algorithms, 2, 93


