# This document summarise the steps to run Pheno on an image

**Preparation of data**

Objective: to prepare a BIL layer stack of fAPAR [floating] and acquisition date [Julday as of IDL, long]

**Extract the area (and smooth it) with SPIRIT**

* First create a reference image for extraction (if already present, skip this point): *File-PseuodImages-Create PI reconversion template* (select a PI image from cid\...\GPI\REF\gpi\_ls1.img, a suitable hdr, one band for the region I want to extract (extract with ENVI from AFR id necessary), and the output template filename)
* Reconvert the PI images*: Time Series – Import Export – Reconvert PI images* (select correct info in scenario). Save in to a directory that may be update when new images come.
* If a MTA metafile is needed use *Basic Tools – Ancillary – Create MTA/VAR image*
* Smooth it: *Basic-Tools- TimeDomain1-Smooth*. Min.unclouded land = 0.0 !!!, swmaxgap=1000

|  |
| --- |
| d:\Users\meronmi\Desktop\Untitled.png |

* Use the metafile to save a bil file (this can be done with: make\_bil\_from\_meta.pro)
* (clean and smooth IDL workspace) Run scale\_bil (main\_clean\_filt) setting relevant info (‘use’ refers to the type of data, fAPAR, blu band, etc; ‘run’ refers to directory settings and file dimensions) twice, for blu band and then for fapar.

If you are doing an update proceed like this:

* Use spirit to uncompress the newest image (from the) last to today
* Update the mta manually and then make the bil go with clean as above

**FOR acquisition day:**

* Extract \*g images as above and do not generate a bil (if it’s a mean/median composite skip this)
* Run generate\_acqJULDAY\_image\_memory (in common\_lib\JD utilities\). This code generates all the JD files and a meta ENVI file pointing to all of them. Same for MODIS, except use mean composite. Per modis attenzione: fa dei JD file che iniziano per vt e non fa il bil, devo prendere il meta e farlo con envi.
* Open the mta in ENVI and make it a bil file

**Compute phenology**

* Run Pheno on the full image or providing a mask (byte image 1/0 indicating where to run the phenol algorithm). On a single processor run **phenot\_batch\_execution,** on the serverrun **decomposer** and then the batch on the server.

If using the “dekad version” results are in progressive dekad from the first of 1997. If using the “day” version, results are expressed in Julian Days.

|  |
| --- |
| How to build quickly the mask:   * Open in ENVI the fapar image you want to use (the spatial extent is important, values are not, can be fapar 8bit unscaled) * Load the political boundary * ROI tool, create the ROI you are interested into and save as a class image (then remove classification from hdr) |

\*If the computer crashes while running there is a chance that the crash did not happen while performing IO operations (in that case the work is lost..). In this case the job can be resumed setting “resume\_from\_save=1” in the code (it checks first if the crash did not happen during IO).

**Realignement**

The problem arises from the following: Phenology refers to recurrent biological events. If a surface is characterized by a recurrent seasonality associated with an arbitrary annual cycle we can study its phenology.

ASSUMPTION: The annual cycle can be characterized by one or two growing seasons. More than two seasons are actually plausible but would be difficult to be detected with available dekadal RS observations.

The proposed algorithm first detect the existence of the seasonality (as measured by autocorrelation or Lomb-Sgarg periodogram), then set the “soft” annual breakpoints of the annual cycle.

Then each annual cycle is analysed independently to extract season characteristics (SOS, EOS, etc.). In one or more annual cycle a season can fail and do not show up.

The output of the algorithm for a given pixel is represented by and ordered and consistent series of season characteristics. For example, considering the feature SOS, the output will be a series of SOS for each of the annual cycle present in the RS time series. SOS can be conveniently expressed as progressive dekads from an arbitrary historic dekad (e.g. from dekad 1 of 1998).

Consistent means that I can only have one (two) season per each annual cycle. Ordered means that the time series of SOS goes from the first to last occurrence.

Studying phenology we may be interest to analyse, for a given geographical area, the timing of the SOS.

We may first be interested in the average dekad at which SOS occurs. This can be computed directly on the pheno output.

In addition, we may be interested in spatial distribution of the timing of a specific SOS, for example those occurring in a given arbitrary year\*. We may thus be interested to analyse, for that year, the anomalies with respect to the average SOS.

\* Note that the calendar year (dekad 1 of January – dekad 3 of December) is just one of the possible 36 annual cycles.

This analysis cannot be operated directly on the output of pheno for two reasons because pheno outputs are ordered and consistent at the pixel level, not among pixels. So the first occurrence of SOS for two different pixels may refer to different annual cycles.

|  |
| --- |
| **Among pixels, pheno vectors can be not aligned!** Consider the following example where the two pixels have a similar annual cycle the SOSs are separated by less than 10 dekads. Just because the RS time series starts at 0 (dek 10 of 1998 for VGT), the first annual cycle is detected for pixel 1 and not for pixel 2. As a result there is almost 1 year of shift between the first occurrence of SOS in the two pixels. So the comparison should not be based on the band number.    The proof that this happen is that if you look to a given band of SOS1 you will find that the range of variation can exceed 36 dekads. |

When targeting a specific vegetation type and a limited geographic area it is usually assumed that the true annual cycle can be a priori defined (assumption: all observed occurrences of SOS fall in such cycle). In such conditions I can extract from the pheno output only the SOS belonging to one specific arbitrary cycle. This is almost equivalent of subjectively setting the annual cycle breakpoints (and not letting them be automatically retrieved by the algorithm). It’s not exactly the same because by arbitrary setting wrong breakpoints may results in impossibility of fitting the model..

This method is not feasible if the above assumption (that all observed occurrences of SOS fall in such cycle) does not hold. This is indeed typically the case when more than one vegetation type is present in the investigated area (each one with its own annual cycle, e.g. grassland and winter crops) and/or the investigated geographic area is large enough to comprise a climatic gradient resulting in a large temporal shift of SOS occurrence for the same vegetation type.

In such conditions, the setting of an hard threshold may result in inconsistent results. For example, a failed season (not displaying any SOS) may be difficult to be detected. Another example is the following, where setting an arbitrary cycle at dekad 15 results in a first year having two growing seasons when the original time series was having only one and consistent growing season per year.





To have a consistent time series we exploit the fact that the output of the algorithm is self-consistent (each pixel has a consistent record of seasons according to its own yearly cycle). That is, the time series is divided into 36 dekads time interval and in each interval a season is searched (may be present or not). Therefore, knowing that season 1 in the arbitrary cycle comes before than season 2, we assign season 1 to the previous annual cycle. Note that this season will then fall in the annual cycle 0 but will have a SOS belonging to annual cycle 1 (it logically belongs to cycle 0 but it was so delayed to fall in the in the next cycle). Another example: the calendar year is requested as cycle of interest. In average SOS1 for a pixel happens at dekoy 33. One year (e.g. 2005) it is delayed until the dekoy 2 of 2006. Still this SOS even if happening in 2006 is assigned to year of interest 2005.

In order to define the correct match between the SOS and the arbitrary annual cycle we use the criterion of minimum distance of SOSs with respect to the midpoint of the arbitrary annual cycle. Consider for example a time series with SOS occurring between dekad 34-36 and 1-3 of the arbitrary cycle.



In such situation we have somehow subjectively decide which SOSs have to be “moved”. For example, the cycle going from 36 to 71 has no SOS, which SOS has to be assigned here?

We exploit here the fact that the SOS time series is ordered by algorithm definition. So there are two possibility for matching it with the midpoint of the arbitrary cycle (18, 54, ..): midpoints align with the previous or with the following SOS. Which one is correct is decided by finding the matching minimizing the sum of square (SOS(i)-midpoint(i)).

*FROM PAPAR IJRS (removed from final version) 3.3. Alignment of phenological variables*

For the analysis of the retrieved phenology it is often convenient to map the timing of a given LSP event (e.g. SOS). For this purpose we first have to distinguish between the pixels showing mono- or bi-modal seasonality. In this latter case we need to produce two maps for each event, referring to its timing for the first and second season of the yearly cycle. After that we can map without ambiguities only the average timing of these events during the period of observation (the RS time series).

However, we are also interested in mapping the timing of an event occurred in a specific time period (observation window, OW), for example the timing of SOS during an arbitrary defined period of one year. We intuitively set this temporal window to one year acknowledging the fact that within this timeframe we expect to observe a single event for each pixel. Nevertheless, in order to produce such maps we have to deal with three issues.

First, the definition of the OW has important consequences on the resulting map. Consider for example an area characterized by a growing season starting between December (dekads 34-36) and January (dekads 1-3). If we set our OW to the calendar year (dekads 1-36) and we are interested in the SOS belonging to year 2010, we may find the timing of two SOS events belonging to two different vegetation cycles: those happened in January 2010 and those happened in December 2010 (belonging indeed to the next growing cycle). If the SOS variability (min-max range) in our area of interest is not covering a full year (as in the example above) we can wisely set our OW to avoid such “border” effect (e.g. setting OW centred on the average timing of the event). Unfortunately, if the geographic extension of our study area is large, the problem cannot be easily solved because the variability in SOS is likely to be large as well, and as we’ll see in point three, it arises and additional problem.

Second, when an OW is set, we may be interested in visualizing other LSP variables that can be either on a circular scale as EOS, MaxT, or on a ratio scale such as MaxV or CUM. In both cases we have to specify the rule for assigning the data to the OW. For example, we set the OW as the calendar year and we want to map MaxV. Since MaxV is by definition referring to a growing season, we can map the MaxV value for those seasons started [ended] in that OW (SOS [EOS] OW). The same logic can be applied to any couple of variables. For example we can map the end of the season of those season started in a specific window of index *m* (OWm). In this case the actual timing of EOS may indeed fall into the next window (OWm+1).

For this purpose we temporally align all the LSP variables of all the pixels according to the selected OW and LSP metric (SOS for example, but it can be any LSP variable on a circular scale). The result of this process is that all LSP variables can be represented by a matrix of dimension (S, L, M) where S and L are the number of samples and lines of the image and M is the number of consequent OWs. In this way, when we are interested in a specific window of index *m* (OW*m*) we can easily retrieve all the associated LSP parameters.

Third, when aligning the LSP variables of a given pixel, we have to keep into account the effect of setting an OW on LSP temporal variability. Consider for simplicity the example of a pixel showing an average SOS corresponding to the second dekad of the calendar year and an OW set to the calendar year. Given the natural temporal variability of SOS, nothing prevents that in a given OW*m* the start may be anticipated of a few dekads, so it will fall in the OW*m*-1. This gives rise to the contradiction that we’ll have two SOSs in OWm-1 (the one of *m-1* plus the anticipated of *m*) and no SOSs in OWm. This example shows that assigning an event to a given OW merely because it falls into it is not a viable method if we cannot be sure to set an OW which is capable to avoid such effect in the whole image.

To solve this inconsistency we adopt the following protocol. By algorithm definition, the output of the LSP retrieval is an ordered sequence of events without gaps (one event per yearly cycle or sub-cycle is recorded, also in the case of season failure) that can be written as the following vector:

(2)

where is expressed in absolute number of dekads from an arbitrary starting date (e.g. the first date of the RS time series), *n* is the cycle number and *N* is the total number of full cycles present for that pixel.

Similarly we can define a vector containing the dates of the first dekad of the selected OW:

(3)

where is the first dekad of the OWs expressed in absolute number of dekad from the arbitrary starting date, *m* is the OW number and *M* is the total number of OWs in the time series (*M* *N*). Note that for a given pixel, *N* may be smaller than *M* because the RS time can start at any time of the actual seasonal cycle.

If, for a given pixel, the OW happen to be well centred on SOS, the will show a one-to-one correspondence to the specified sequence, and we just have to find the vector index *i* for which

(4)

If this is not the case, not all SOSs fall in the sequence of OWs (there is no *i* satisfying Eq. 4). The criterion we adopt to align the selected events is to minimize the overall distance between the SOSs and the midpoints of the OWs. This is achieved by minimizing the following cost function with respect to *i*:

(5)

where is the midpoint of OWx+1. In this way we consistently get only one event for each OW with an actual timing does not necessarily falls in that OW.

Use **A0\_TZP\_realign\_pheno\_products\_JD.pro** to realign the bands to a prescribed 365 days cycle. This cycle is defined by the starting DOY (frstDOYofCycle). If frstDOYofCycle = 1 it is the calendar year.

NOTE that:

1. The alignment performed on the SOS (both 1 and 2) so that in a given pixel SOS1 will be the first season occurring with respect to the prescribed year cycle (this may implies an exchange of phenol output SOS1 and 2).
2. Since the alignment is performed on a PE (Pheno Event, e.g., SOS), it may happen that another aligned PE (e.g., EOS) does not fall in the same cycle of SOS (normally in the next).
3. None of the pheno product values are changed by this procedure which just align the pheno output bands.

**Compute deviations and anomalies**

* For a general overview of the results, transform aligned circular pheno indicator (sos, eos, maxt) from absolute decade (0 is the first dekad of the fAPAR time series) to relative dekad with respect to the calendar year (or given cycle starting at any DOY) using **TZP\_imageJD\_2\_imageDOC.pro** \*
* Treat season failures first with failure\_analysis that, at the department level computes the number of pixels showing failure for each year an return the devY (% variation with respect to mean computed over all year, excluding the year of interest)
* Use **a1\_anomaly\_analysis\_JD** which calls **TZP\_pheno\_anomalies\_JD** and compute the required anomaly or delta and then, for a given band the user select (boi, band of interest in the phenol product), produces department level averaged maps separately for those regions having 1 growing season per year (only map of anomaly) and 2 ngspy (two maps of anomaly, one for the first season and one for the second season). Anomalies are computed at the pixel level and the averaged at the department level. On the other hand, for what concern season failure, everything is done at the department level: for each department, the number of pixels showing season failure is computed for each the available year (excluding incomplete years, see variable ‘bex’), and then anomaly % variation is computed, always at the department level, for that particular year (failure\_analysis).

Note that:

pheno\_anomalies works pixel by pixel, so pheno products MUST be aligned to spatially interpret the data afterwards

Treatment of season failure

Season failures data (-999) are normally kept during processing (i.e., realignment, JD to DOC ). In **TZP\_pheno\_anomalies\_JD** season failure (-999) is not used to compute mean and it is kept in output if val999 is set to NaN. Be aware that if val999 is set to 0 it will be considered in subsequent mean operations. So be aware that, for ‘min’ you will get the minimum of a given variable excluding crop failures, that have to be treated separately (see also below)

\*TZP\_imageJD\_2\_imageDOC.pro

Transform a PE file (expressed in JD) to:

Relative DOC (Day Of Cycle) with respect to the required cycle starting at *frstDOYofCycle* (so a value of 1 will result in the calendar year cycle).

Note: The DOC is always relative. If for example the required cycle is the calendar year and sos1 occurs at 36 of the previous year), it will be registered as -1. In fact, output is not circular, is TZP.

The anomaly analysis has to be performed onDOC

\*\* Anomalies, use TZPpheno\_anamolies which requires TZP data or dekoc data (consider using option reliability):

* dlt: The delta from avg [dM(i,j,t)=M(i,j,t) - Mean(M(i,j,\*))]. Avg is the mean of all avaible year
* dltY: The delta from of every year from avg computed over all years excluding Y
* dev: The anomaly from avg [ dM(i,j,t)=M(i,j,t) / Mean(M(i,j,\*)) \*100]
* devY: The anomaly from of every year from avg computed over all years excluding Y
* Min or max: DO NOT USE WITH dekoy data. The year of min [max] for pixel (expressed as number of bands counting from 1), failures (-999) are not considered, output will be NaN if all the years are failure. If there is more than one identical min or max, the most recent is recorded. So they may be not absolute max or min

**ARCGIS**

IDL files must be saved as “.hdr” and “.img”. They should not start with a number. Copy map info into hdr file.

Somalia, comments

Processing:

* Pheno
* Realign
* Failure analysis
* Dekoc (sos1 and sos2)
* Anomaly\_analysis. *dltY*: (with dekoc option) A19-1997\_sos1\_dekoc, (with dekoc option) A19-1997\_sos2\_dekoc, A19-1997\_len; *devY*: A19-1997\_len, A19-1997\_acc. To get the avg you have to use the TZP anomalies.

Ngspy: Number of growing season per year (possible values: 0, 1, 2)

Season failures: % deviation from the man number pixels showing season failure in a Gaul. Separate bands for areas with 1 GSPY (1 band of failure) and 2 GSPY (2 bands of failures)

**Then, referring only to those pixels showing no failure, there are the anomalies:**

* DLTY (delta compared to the mean all years excluding that of interest), always 3 bands (1GSPY\_seson1, 2GSPY\_seson1, 2GSPY\_seson2): SOS, LEN
* DEVY (% with respect to the mean all years excluding that of interest), always 3 bands: LEN, ACC

GIS:

* Ngspy
* Failures
* SOS anomalies
* LEN anomalies
* ACC anomalies

In a previous study (Meroni et al., 2012?) we used the autocorrelogram of the FAPAR profile to determine the number of growing season per year. However, this analysis failed to detect the correct seasonality when the FAPAR time series showed poor periodicity (i.e., lack of a dominant FAPAR evolution pattern throughout the years) or predominance of one of the two growing seasons.

In the present study we exploited the Lomb normalized periodogram (Scargle, 1982) to determine the power spectrum of unevenly spaced data. More specifically we used the ratio between the power associated to a single and double yearly frequencies. Values below [above] one indicate that the power of the mono-modal component is greater [smaller] than that of the bi-modal one. Values greater than one indicate the presence of a secondary cycle that may be associated with a true secondary season or with noise exhibiting a repetitive pattern in the time series. A pragmatic approach based on a threshold is here used to discriminate between these occurrences. The threshold value was empirically set to 6 using two criteria. First we visually inspected a number of profiles representative of diverse environments in the study area, ranging from clearly mono- and bi-modal seasonality to more challenging situation such as locations expected to be mono-modal but showing a small secondary bump and locations characterized by a discontinuous manifestation of a second season. This analysis showed that ratio values below the threshold are usually associated with bi-modal seasonality while values above are mono-modal. The threshold was also effective in distinguishing mono-modal season affected by a regular noise pattern. Second we analysed the spatial patterns of the ratio in the whole study area to gain confidence that the selected threshold would result in a seasonality map that regionally match the expected pattern.

However, some locations characterized by a regular noise pattern between the seasons may still provide a ratio value exceeding the threshold if the main annual peak is very pronounced. This ambiguity is solved exploiting the fact that in this case the amplitude of the secondary cycle is very low and the optimization is not even attempted. Therefore, the number of cycles is initially set to two and the time series is processed. After that, if no season were found in the secondary cycle, the number of cycle forced to one and the processing restarted.

SHOW FIGURES? (see lomb rat.ppt)

A significance test on the power peaks is possible but unable to provide a good discrimination power because mostly detecting two significant cycles. A more reliable test would include the quantification of the uncertainty in the observed time series in the test. This would allow to reduce the weight of uncertain observations in the computation of the periodogram..?

References

Scargle, J. D. (1982), Studies in astronomical time series analysis. II - Statistical aspects of spectral analysis of unevenly spaced data. Astrophysical Journal 263: 835.