



# omflatfield

January 27, 2025

## Abstract

Creates a tracking shifted flatfield and applies it to an OM OSW image.

## 1 Instruments/Modes

Instrument	Mode
OM	FAST
OM	IMAGING

## 2 Use

pipeline processing	yes
interactive analysis	yes

## 3 Description

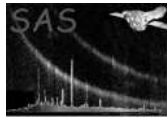
This task creates a tracking shifted flatfield and applies it to an OM OSW Image.

The tracking shifted flatfield is created by extracting a window from the OM In Orbit Flatfield. This window is made slightly bigger than the OM OSW Image to accomodate the tracking shifts of pixels within the In Orbit Flatfield. The window is also oversampled so that incremental tracking shifts may be applied. The resulting oversampled tracking shifted flatfield is rebinned and resized before it is divided into the OM OSW Image and output as a PPS product. The flatfielded OM OSW Image is also output for use in subsequent pipeline tasks.

## 4 Parameters

This section documents the parameters recognized by this task (if any).

Parameter	Mand	Type	Default	Constraints
samplefactor	no	integer	1	1-10



Spatial sampling factor for tracking shifts

<b>set</b>	yes	string	none	
------------	-----	--------	------	--

OM Image filename

<b>thxset</b>	yes	string	none	
---------------	-----	--------	------	--

OM Tracking History Data Auxiliary filename

<b>inorbitflatset</b>	yes	string	none	
-----------------------	-----	--------	------	--

OM In Orbit Flatfield filename

<b>tsflatset</b>	no	string	none	
------------------	----	--------	------	--

OM Flatfield filename

<b>outset</b>	yes	string	none	
---------------	-----	--------	------	--

OM Flatfielded Image filename

## 5 Errors

This section documents warnings and errors generated by this task (if any). Note that warnings and errors can also be generated in the SAS infrastructure libraries, in which case they would not be documented here. Refer to the index of all errors and warnings available in the HTML version of the SAS documentation.

### **allocError** (*fatal*)

Can't allocate memory for internal array

### **badFileMode** (*fatal*)

Invalid datamode <dataMode> in set <set>

### **ZERODRIFT** (*warning*)

OM tracking history is not present for this exposure so no tracking shifts can be applied

*corrective action:* No tracking correction will be applied

### **deAllocError** (*warning*)

Failed to release memory for <name>

*corrective action:*

### **zeroValuedFlat** (*warning*)

One or more elements of the tracking-shifted flat field (usually at the edge) were unexpectedly found to be zero. This is likely to be a sampling effect.

*corrective action:* Resets zero valued pixels to unity and sets a non-zero value in the quality array

### **Uniform flatfield** (*warning*)

No flatfield correction will be applied to the image

*corrective action:* The OM flatfield is currently set to unity (see *omflatgen* task documentation) so this warning always appears. This may change in the future. The use of a uniform flatfield is 'unconventional', hence the warning (notification) to the user.



## 6 Input Files

1. Intermediate OM Image file (taken from OMCOSFLAG).
2. Intermediate OM Tracking History Data Auxiliary file (taken from OMPREP).
3. Intermediate OM In Orbit Flatfield file.

## 7 Output Files

1. OM PPS Flatfield file.
2. OM Flatfielded Image file (for input in to OMMODMAP).

## 8 Algorithm

subroutine omflatfield

    read in task parameters

    open OM OSW Image file

    determine file type from datamode in primary header

    read in the science window keywords from the primary header

    calculate image binning

    extract window data from MODES binary table extension

    extract OM OSW Image

    open OM Tracking History Data Auxiliary file

    determine file type from datamode in primary header

    extract tracking history data from OMTHX1 binary table extension

    check for poor tracking solution and warn

    extract OM In Orbit Flatfield file

    calculate maximum drift in x and y

    extract localised in orbit flatfield

    oversample localised in orbit flatfield

    loop over tracking frames to construct oversampled tracking shifted flatfield



```
    loop over discretised tracking steps

        add shifted oversampled localised in orbit flatfield to oversampled tracking shifted flatfi

    end loop

end loop

normalise the summed flatfield

resize oversampled tracking shifted flatfield

divide OM OSW Image by tracking shifted flatfield

resize and rebin OM OSW Image with appropriate binning factors

create output OM OSW Image file

create output OM PPS flatfield

end subroutine omflatfield
```

## 9 Comments

Since no proper flatfields have been made so far, a check is made to see if all the flatfield pixels are unity- if so the input image is not corrected but a warning issued. This saves a considerable amount of computing time.

## 10 Future developments

## References