

SIXTE: Docoding the XGIS mask

Philipp Thalhammer

December 27, 2025

1 Introduction

The upcoming THESEUS satellite will be equipped with two main high-energy X-ray and γ -ray detectors. One of which is the coded mask instrument *XGIS* (1). It is sensitive to both X-ray and γ -rays in the 2 keV–10 MeV range. It consists of a coded mask and a silicon drift detector layer. Some relevant geometrical parameters of the coded-mask and detector geometry are listed in Table 1. In this document we will briefly outline the *SIXTE* simulation setup for the *XGIS* detector in its current state and the deconvolution algorithm that is currently available for its output. This document also provides a brief tutorial on how to use the deconvolution script.

This document explains the deconstruction algorithm implemented as part of *SIXTE* to recover the sky image from the shadow patter on the detector plane. The layout of the *XGIS* detector as used by *SIXTE* is shown in Fig. 1. It comprises 10×10 individual detector modules with 8×8 pixels each, with a 1 mm gap between individual modules. In the current version the *SIXTE* version uses a simplified version of the absorption physics within the mask material - considering the mask geometrically thin and fully opaque below 150 keV (the X-mode) and fully transparent above 150 keV (the S-mode). The two energy ranges are therefore treated as separated detectors defined through different xml files.

2 The Projection:

To project a single photon form a position in the sky onto the detector the shadogram is shifted by

$$\delta x = d_{\text{mask}} \tan(\delta\phi). \quad (1)$$

Then a random x-/y-position on the mask is sampled and if his position corresponds to an open position on the mask, and lies on the detector, the impact position on the detector is forwarded to the following modules of *SIXTE*. The photon energies are being sampled from according to the ϕ and θ dependent RMFs, while the off-axis effects generated from the mask effects are being accounted for via vignetting as derived from the off-axis ARFs. The detector readout is handled via the respective *SIXTE* module for a SDD detector.

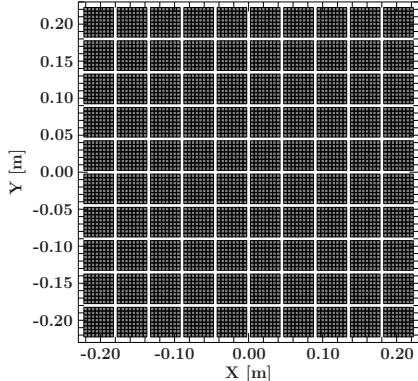


Figure 1: The layout of the *XGIS* detector as used by *SIXTE*.

Mask	
Width	56.1 cm
Mask distance	63.0 cm
Pixel width	1 mm
Resolution	55x55
Thickness	1 mm
Material	Tungsten (W)

Detector	
Total width	45 cm
Pixel width	0.5 mm
Detector setup	10x10
Pixels per detector	8x8

Table 1: Relevant parameters of the *XGIS* detector

3 The Deprojection:

This results in each source throwing a shadow pattern on the detector - the so called shadowgram. Deconstruction of the original sky pattern requires to calculate the cross-correlation between the detector image and the shifted mask pattern for each position in the sky. This can be calculated by summation of element wise matrix multiplication.

$$S = D * G = \sum_{ij}^{N_{\text{pix}}} D_{k,l} G_{i+k,j+l} \quad (2)$$

Here the matrix G corresponds to the mask pattern rescaled to provide a correctly normalized devolution that is one average zero for an empty region of the sky. It is, however, much more computationally efficient to calculate the convolution via a multiplication in Fourier space.

$$S = \mathcal{F}^{-1} (\mathcal{F}(G) \cdot \mathcal{F}(D)) \quad (3)$$

For a exhaustive overview of the deconvolution algorithm, we refer to (**author?**) (2, and references therein).

4 Implementation:

The implementation of the deconstruction algorithm is done in Python using common libraries such as `numpy` and `scipy` and can be called via `deconvolve.py`. For more information `deconvolve.py --help` can be called.

5 Example:

In the following we illustrate an example run including both the projection and the deprojection. Before a single off-axis source is defined in a respective simput file, the simulation is run via *sixtesim*:

```
1 $SIXTE/bin/sixtesim \
3   XMLfile=${xmlfile}/xgis-x.xml \
4     RA=0.0 Dec=0.0 \
5     Exposure=1000 \
6     Simput="${simputdir}/single_source.fits" \
7     Prefix="output/" \
8     EvtFile=test.fits \
9     Background=y \
clobber=y
```

As the detector array of *XGIS* consists of 10×10 individual detector arrays, *SIXTE* will provide an event-file for each individual detector. As a user this is likely inconvenient, and it is therefore recommended merging individual event-files by running a script similar to the following:

```
1 ls test*.fits > evts.lst
2 fmerge infiles=@evts.lst \
3           outfile=test_merged.fits columns=-
```

The merged event-file can then be used for the deconvolution and reconstruction of the sky image. For this a python3 script can be provided, which requires the source position for the generation of spectra and light-curves and prefix for the output files names. Default ARF and RMF paths are assumed but mostly likely specific paths to need to be supplied as input parameters. A minimal example can be seen below:

```
1 ./deconvolve.py
2   --ra_src 250 --dec_src -50
3   --outfile "test"
--infile "test_merged.fits"
```

Running *SIXTE* results in the detector and reconstructed sky image show in Fig. 2. For a more realistic representation of *XGIS*, we simulate a pointing towards the galactic center based on the INTEGRAL reference catalog. The detector and sky image are shown in the Fig. 3.

The tool *deconvolve.py* also generates light-curves and spectra for a specific sources position. For this the deconvolution is performed for each spectral channel and *nbins* time bins. Appropriate header keywords pointing towards the response files are set, such that the pha-file can be used an any spectral fitting software, such as ISIS or XSPEC. Systematic uncertainties of a few percent

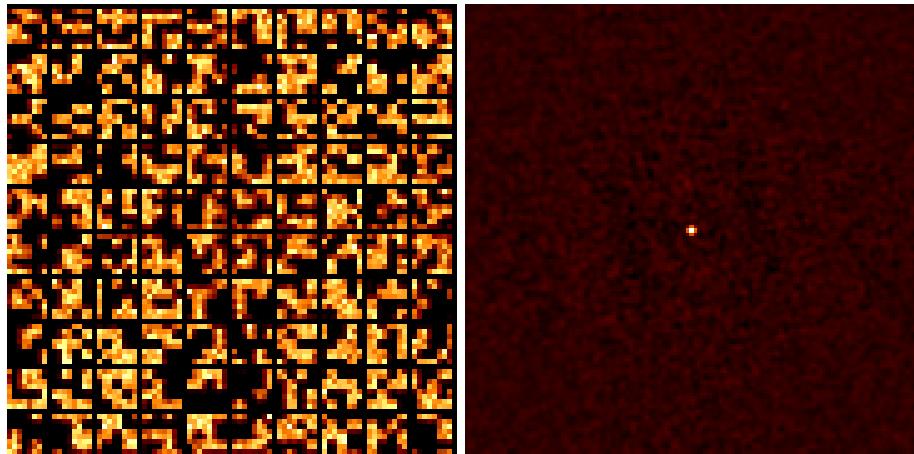


Figure 2: The detector image (left) and the reconstructed sky image (right) for a simulation with a single off-axis source.

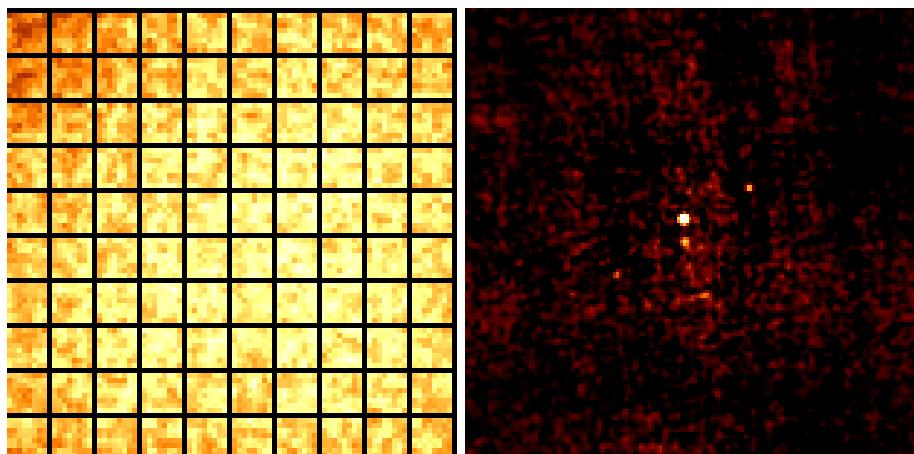


Figure 3: The detector image (left) and the reconstructed sky image (right) for a simulation of the galactic center. The detector image shows the shadow pattern cast by the coded mask, while the sky image shows the reconstructed source distribution.

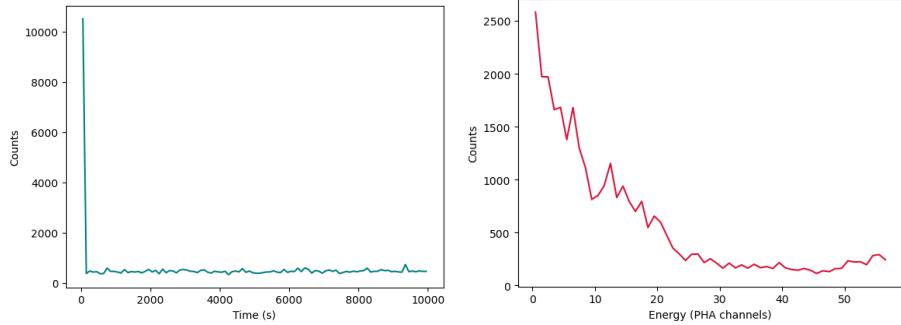


Figure 4: Light-curve and spectrum for a simulated GRB at the very beginning of a 10 ks observation

might be needed for satisfactory fits. Since the reconstruction algorithm has to run many times for spectra and light-curve extraction this is the slowest part of the script. For a faster generation of sky images the keywords *no_spe* and *no_lc* can be used. The final image will be saves as a FITS file and supplies with WCS coordinates. In the current iteration this step assumes that the roll angle of the telescope is aligned with the equatorial coordinate grid, i.e., x-z coordinates are aligned with RA-DEC. Fig. 4 shows the extracted light-curve and spectrum for a simulated GRB with a flux of 10^{-6} erg/s/cm².

References

- [2] Goldwurm, A. & Gros, A. 2022, Coded Mask Instruments for Gamma-Ray Astronomy, 15
- [1] Labanti, C., Amati, L., Frontera, F., et al. 2021, The X/Gamma-ray Imaging Spectrometer (XGIS) on-Board THESEUS: Design, Main Characteristics, and Concept of Operation