

Highlights

FITrig: A High-Performance Detection Technique for Efficient Ultra-Long-Period Pulsars

Xiaotong Li, Karel Adámek, Wesley Armour

- **tLISI — a novel transient-oriented IQA index:** A statistically-grounded IQA metric specially designed to highlight pulsar-specific variations through probability theory, enabling more reliable transient detection.
- **Elimination of deconvolution:** FITrig completely removes the need for deconvolution in the detection process while maintaining accuracy, significantly reducing computational overhead.
- **Dual-domain detection framework:** Two branches of FITrig (image domain and image-frequency domain approaches) are developed, significantly improving the capability to detect faint ULP pulsars.
- **GPU-powered real-time processing:** FITrig is highly parallelised with the GPU implementation, enabling real-time processing and scalability for large datasets.

FITrig: A High-Performance Detection Technique for Efficient Ultra-Long-Period Pulsars

Xiaotong Li, Karel Adámek, Wesley Armour

Oxford e-Research Centre, Department of Engineering Science, University of Oxford, 7 Keble Rd, Oxford, OX1 3QG, UK

Abstract

Ultra-long-period (ULP) pulsars, a newly identified class of celestial transients, offer unique insights into astrophysics, though very few have been detected to date. In radio astronomy, most time-domain detection methods cannot find these pulsars, and current image-based detection approaches still face challenges, including low sensitivity, high false positive rate, and low computational efficiency. In this article, we develop Fast Imaging Trigger (FITrig), a GPU-accelerated, statistics-based method for ULP pulsar detection and localisation. FITrig includes two complementary approaches — an image domain and an image-frequency domain strategy. FITrig offers advantages by increasing sensitivity to faint pulsars, suppressing false positives (from noise, processing artefacts, or steady sources), and improving search efficiency in large-scale wide-field images. Compared to the state-of-the-art source finder SOFIA 2, FITrig increases the detection speed by 4.3 times for large images ($50\text{K} \times 50\text{K}$ pixels) and reduces false positives by up to 858.8 times (at 6σ significance) for the image domain branch, while the image-frequency domain branch suppresses false positives even further. FITrig maintains the capability to detect pulsars that are 20 times fainter than surrounding steady features, even under critical Nyquist sampling conditions. In this article, the performance of FITrig is demonstrated using both real-world data (MeerKAT observations of PSR J0901-4046) and simulated datasets based on MeerKAT and SKA Array Assembly (AA) 2 telescope configurations. With its real-time processing capabilities and scalability, FITrig is a promising tool for next-generation telescopes, such as the SKA, with the potential to uncover hidden ULP pulsars.

Keywords:

1. Introduction

Searching for pulsars (Lorimer and Kramer, 2004) is a popular topic in radio astronomy since detecting and observing their radio emissions allows astronomers to test theories such as general relativity (Hoffmann, 1968; Desvignes et al., 2019) and to study astrophysical processes governed by magnetohydrodynamics (MHD) (Urpin, 2011; Okamoto and Sigalo, 2006; Tchekhovskoy et al., 2013). Ultra-Long-Period (ULP) Pulsars, such as PSR J0901-4046 with its exceptionally long 76-second rotation period (Caleb et al., 2022), are a unique class of pulsars characterised by unusually slow spins. ULP pulsars could reveal hidden neutron star populations and improve gravitational wave studies, while their emission properties also challenge traditional models of neutron star physics (Bezuidenhout et al., 2025).

However, ULP pulsars introduce significant detection challenges, as their extremely slow spin rates produce faint, widely spaced radio pulses that are often masked by noise. Even harder, their long-period emission and transient nature demand prolonged observations to reliably pinpoint true signals from false positives. These challenges explain why most pulsar detection techniques, which operate primarily in the time-frequency domain (Lorimer and Kramer, 2004; Ransom et al., 2002; Andersen and Ransom, 2018), struggle to identify ULP pulsars.

Transitioning pulsar search from the time-frequency domain to the image domain enables the detection of ULP pulsars because their long-period signals form coherent patterns (e.g., repeating bright spots) that are more easily distinguished from noise than widely spaced peaks in traditional periodicity searches. Fast Imaging (FI) reconstructs images from visibilities collected over short timescales and identifies transient candidates from these images (referred to as “snapshots”) through statistical selection (Trott et al., 2011). Recent astrophysical results from image-based transient searches include the detection of binary white dwarf pulsars (Pelisoli et al., 2023) and ULP neutron stars (Caleb et al., 2022) in radio bands. This area has motivated detection tools such as *realfast* (Law et al., 2018) for the Karl G.

| Aspect | Time Domain Detection | Image Domain Detection |
|----------------------|--|--|
| Event types detected | Ideal for high-frequency pulsars. | Better for low-frequency pulsars. |
| Temporal resolution | Excellent temporal resolution. | Limited by image integration time. |
| Spatial resolution | Poor spatial resolution. | High spatial resolution. |
| Computational demand | Lower; processing focused on time-series analysis. | High; imaging involves Fourier transforms and deconvolution. |
| Data volume | Relatively smaller data volumes. | Larger data volumes. |

Table 1: Comparisons of time-domain vs. image-domain pulsar detection.

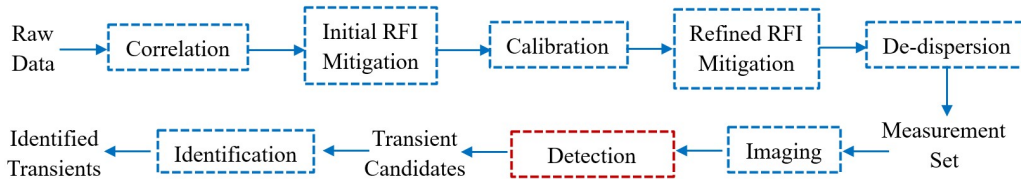


Figure 1: Schematic representation of an image-based radio transient detection workflow, where the highlighted “Detection” component indicates the primary focus of this article.

Jansky Very Large Array ¹ (VLA; Thompson et al. 1980). Table 1 presents a comprehensive analysis that compares the benefits and limitations of pulsar detection in the image domain against the time domain. An example workflow for image-based radio pulsar detection is illustrated in Fig. 1 (Burke et al., 2019; Thompson et al., 2017).

In the FI workflow, after creating wide-field images for each time slot, pulsars are localised in the resulting snapshots using source-finding techniques, such as the SOurce FInding Application (SOFIA) ² (Westmeier et al., 2021), SFIND (Hopkins et al., 2002), AEGEAN (Hancock et al., 2012), and Python Blob Detector and Source Finder ² (PyBDSF). Among them is SOFIA 2, a parallel code that utilises OpenMP for multi-threading in time-critical parts

¹<https://public.nrao.edu/telescopes/vla/>

²<https://www.ascl.net/1502.007>

of the implementation. In it, the input image is smoothed across many spatial and spectral scales to pinpoint notable emissions. This algorithm is known as the Smooth and Clip (S + C) finder (Serra et al., 2012).

However, challenges remain in current source finders for ULP pulsar detection:

- **Sensitivity:** The detection of pulsars, particularly for ULP pulsars, is often limited by their low signal-to-noise ratio (SNR) in radio astronomical images.
- **False Positive Rate:** Robust separation of true pulsar signals from artefacts is critical to avoid misclassifying false positives as real ULP pulsars.
- **Computational Efficiency:** The growing volume of data from next-generation telescopes demands more efficient pulsar finding algorithms. A computationally optimised ULP detection technique is essential for real-time analysis, enabling timely follow-up observations of candidate pulsars.

To address these challenges, we develop a new detection technique for ULP pulsars, called Fast Imaging Trigger (FITrig). FITrig achieves high sensitivity through a target-sensitive image assessment method, tLISI (transient-oriented Low-Information Similarity Index), proposed in this work. False positives are minimised by a statistic-based auto-selection process integrated into the triggering workflow. In addition, high computational efficiency is achieved through the GPU-accelerated design of FITrig, which leverages its highly parallel architecture to rapidly process large datasets and enable timely identification of candidate pulsars.

To support clarity throughout this article, some key terms are briefly explained here. A “celestial transient”, or simply “transient”, refers to a radio source whose emission varies noticeably over time in observations. A “steady source”, in contrast, denotes a radio source whose emission remains largely constant, showing no significant variability during observations. The “dirty beam” is the point spread function (PSF) of the telescope array resulting from its incomplete uv -coverage; we use the term “dirty beam” throughout. Finally, “steady features” refer to the structures formed by steady sources together with the dirty beams convolved on them. For reference, a table summarising the technical terms and their abbreviations used in this article is provided in Table A.5.

The methodology of FITrig is proposed in Section 2. Its performance is demonstrated by the experimental results in Section 3, starting with the performance of FITrig on the real dataset of PSR J0901-4046 measured by MeerKAT (Booth and Jonas, 2012; Davidson, 2012; Jonas, 2009). We then test FITrig on more challenging cases using simulated datasets covering various pulsar brightness levels and frequencies, based on the telescope layouts of MeerKAT and Square Kilometre Array (SKA; Dewdney et al. 2013) AA2³. This further illustrates its performance in terms of computational efficiency (Section 3.2), sensitivity (Section 3.3), and false positive rate (Section 3.4). Section 4 presents the conclusions.

2. Method

2.1. Problem Analysis

In radio interferometric imaging, the relationship between visibilities and the received sky brightness distribution (SBD) (Burke et al., 2019) is given by

$$V(u, v, w) = \iint I(l, m) e^{-i2\pi(ul+vm+w(\sqrt{1-l^2-m^2}-1))} dl dm, \quad (1)$$

where V represents the observed visibilities, I represents the received SBD, and l, m, n and u, v, w are coordinates in the spatial and frequency domains, respectively. The image obtained by inverse Fourier transforming the observed visibilities is called the dirty image, where celestial sources appear convolved with the dirty beam.

In image-based pulsar detection, the influence of the dirty beam is mitigated through either deconvolution techniques (Pelisoli et al., 2023) such as CLEAN (Hogbom, 1974) and Maximum Entropy Method (MEM; Cornwell and Evans 1985), or by computing difference images from time-adjacent snapshots. While deconvolved images yield higher detection accuracy, they require significantly more processing time. Conversely, difference images offer a faster alternative but at the cost of increased false positives and reduced sensitivity to fainter pulsars as a result of their higher noise level compared to deconvolved images.

³<https://www.skao.int/en/news/174/green-light-given-construction-worlds-largest-radio-telescope-arrays>

From the perspective of computational efficiency, the latter is a better choice. However, there are some issues remaining with image-based ULP pulsar searches.

- **Localised Search Regions:** When conducting wide-field transient searches that prioritise large-area coverage, pulsars often occupy only a small proportion of the image, making whole-image searches computationally wasteful.
- **Sparse Temporal Appearance:** ULP pulsars appear in just a few snapshots within time-sequential difference images. Blindly searching every difference frame wastes resources, especially for the large volume of data collected by next-generation telescopes.
- **Dynamic “Steady” Features:** Ideally, difference images would contain only the pulsar signal convolved with the dirty beam, as steady sources and noise would cancel out when subtracting adjacent snapshots. In reality, the dirty beam changes with Earth’s rotation and steady sources fluctuate in time, resulting in visual artefacts due to partial cancellation of the dirty beam in the difference image. These artefacts, when bright enough, dominate the difference image and obscure the pulsar signal.

To address these issues, we propose a method, FITrig, tailored for detecting ULP pulsars in difference images from dirty snapshots.

2.2. Transient-oriented Low-Information Similarity Index (tLISI)

In order to avoid whole-image searches, FITrig is designed to start by localising potential regions that contain pulsars within the SBD. Intensity-sensitive Image Quality Assessment (IQA) methods have been developed to pinpoint potential regions containing transients (Li et al., 2024a), allowing an efficient search of the entire image. In this context, while these approaches are conventionally termed IQAs, they function not as quality assessors, but rather as detectors. Among them, Low-Information Similarity Index (LISI; Li and Armour 2022) is suitable for assessing images with point-like sources, while augmented LISI (augLISI; Li et al. 2024b) is better for assessing images with extended structures. Aiming to process radio astronomical images, both LISI and augLISI are designed to be more sensitive to high-intensity

differences in the two input images being compared, but less sensitive to low-intensity differences.

However, in the detection of ULP pulsars, high-sensitivity detection requires the detector to find fainter transients that may be obscured by dynamic steady features (as analysed in Section 2.1) and noise. If an IQA method is less sensitive to low-intensity differences, those faint transients would be ignored. Therefore, in this article, we propose transient-oriented Low-Information Similarity Index (tLISI), a new IQA method for detecting ULP pulsars in radio astronomical images.

2.2.1. ULP Pulsar Detection in Consecutive Images

As analysed in Section 2.1, FITrig (starting from tLISI) is designed to operate on the difference images between adjacent snapshots, as shown in Fig. 2. The example figures were simulated by Oxford’s Square Kilometre Array Radio-telescope simulator (OSKAR; Dulwich et al. 2022). Figure 2 (a) and (b) show two crops from time adjacent snapshots, with the pulsar present only in Snapshot 2 of crop (a), highlighted by the red circle. In (b), the first two difference images show shadowed shapes, which are parts of the side lobe of the convolved dirty beam, but their intensities are much lower than those of the pulsar (as shown in the first two difference images in (a)), as expected.

Typically, a ULP pulsar has a period longer than twice the sampling period of the telescope. Given their sparse temporal appearance, a detection model is built on three-snapshot units, where the ULP pulsar appears in only one snapshot, as shown in Fig. 3. The three highlighted images represent the input for tLISI when comparing the two difference images. Both snapshots and difference images are divided into tiles to localise potential region and enable parallel computing. All tiles at the same position in the time-sequential images need to have the same shape, whereas the tile shape can be regular or irregular. To reduce computational costs, it is preferable to use square or rectangular tiles. The choice of tile size affects the sensitivity of pulsar detection, which will be discussed in Section 3.

The tLISI metric is designed such that its value is higher when both difference images include a pulsar, or neither image contains a pulsar. On the other hand, the tLISI value is lower when one difference image includes a pulsar while the other does not. In our design, tLISI ($tLISI \in [0, 1]$) equals 1 when the two difference images are exactly the same and 0 when the two difference images are very different. Designed for parallel computing

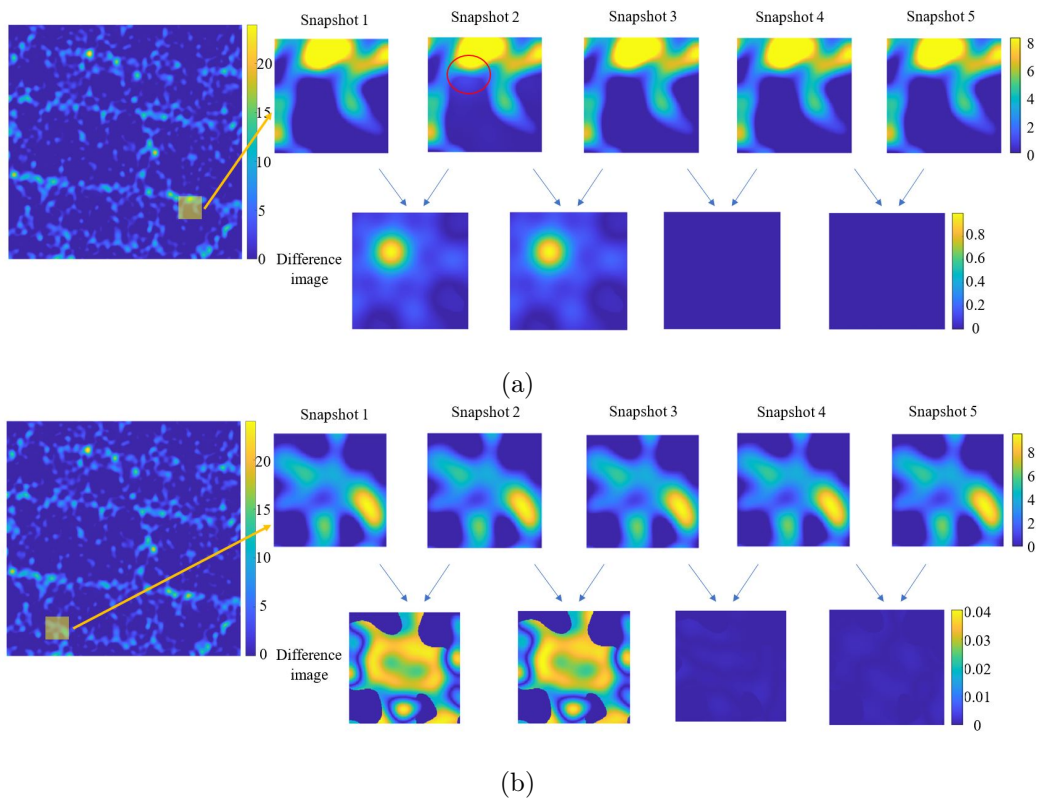


Figure 2: Example snapshots and their difference images reconstructed from a simulated dataset generated by OSKAR using VLA D array layout.

efficiency, the tLISI metric processes each tile independently.

2.2.2. Side Lobes of Dirty Beam

While pulsar snapshots (images containing the pulsar) can be identified by searching for difference images that include the dirty beam convolved with the pulsar, tLISI must accurately localise “pulsar tiles” (those containing the true pulsar signal) while rejecting tiles that only contain side-lobe artefacts from the dirty beam. In other words, tLISI needs to be more sensitive to tiles with larger intensity differences between adjacent snapshots. However, what is the proper metric to define “intensity differences”?

When the dirty beam’s side lobes extend over a wide area and the main lobe is narrow, “non-pulsar tiles” (those without any pulsar) may still exhibit a high sum of pixel intensities, potentially even greater than that of the pulsar tile. For example, Fig. 4 illustrates this effect: a non-pulsar tile (Fig. 4 (a))

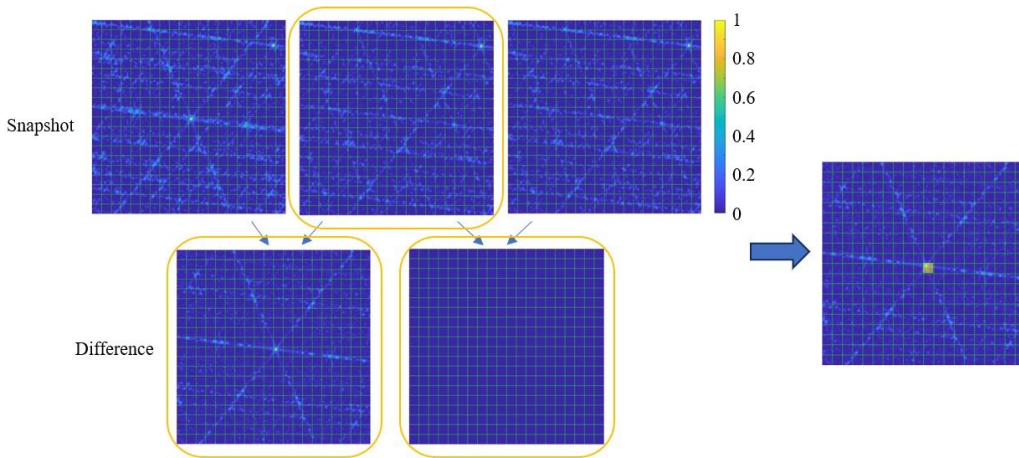


Figure 3: Three-snapshot unit in the detection model. These images are reconstructed from a simulated dataset generated by OSKAR using the VLA D telescope layout. Suppose the index of the top-left tile is (1,1). In this example, the pulsar is primarily located in tile (11,11). The colour bar indicates the intensity scales, which are normalised in this example for illustration purposes. In the right figure, the yellow-shaded tile indicates the tile selected by tLISI, which will be passed to the next step for further localisation.

can show a significant total pixel intensity from side lobes, even though its maximum pixel intensity remains lower than that of the true pulsar tile (Fig. 4 (b)).

Therefore, to minimise the false positive rate, tLISI should consider both the sum of pixel intensities and the maximum intensity within a tile, ensuring robust discrimination between pulsar tiles and side lobe artefacts.

2.2.3. Dynamic Steady Features Mitigation

As shown in the example in Fig. 5, dynamic steady features can produce intensity differences (from dirty beam convolved on bright steady sources) that surpass the pulsar flux in adjacent difference images. As seen in the figure, crop (a) contains very bright steady features, resulting in significant changes in the difference images, whereas the changes in crop (b) are smaller, even though it contains the true pulsar.

To assess the significance of these changes quantitatively, we define a ratio r of intensity variation to the corresponding pixel intensity in the snapshot; the mathematical formulation is presented in Section 2.2.4. We apply a threshold of 1 to the ratio r by $r \rightarrow \min(r, 1)$, ensuring that:

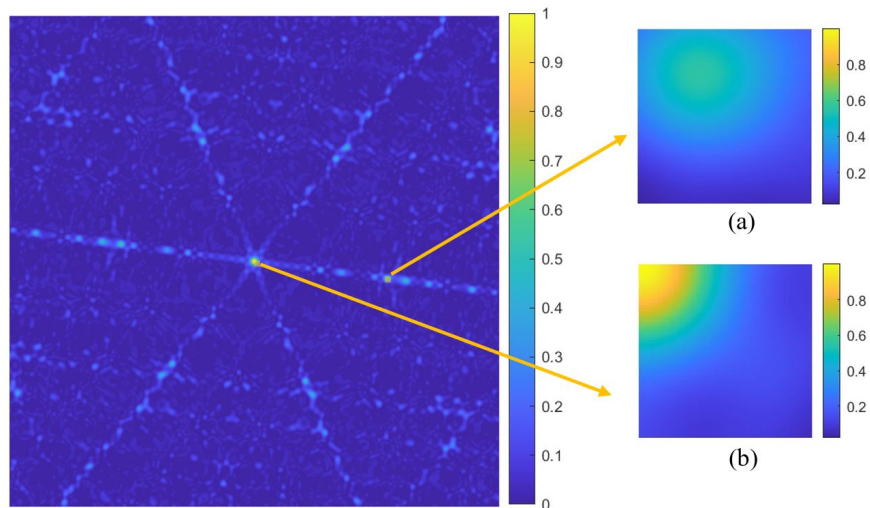


Figure 4: An example of tiles in a difference image showing the side lobe effect of dirty beam: tile (a), which contains no pulsar, shows a larger sum of intensities due to the side lobes of the dirty beam convolved with the pulsar, compared to tile (b), which does contain a pulsar. This example is simulated using the VLA telescope layout.

- For pixels with low reference intensity but significant changes, the sensitivity of tLISI remains unaffected;
- For pixels with high reference intensity, the sensitivity of tLISI is reduced.

This thresholded ratio approach effectively reduces the sensitivity of tLISI to variations in steady features while preserving its sensitivity to pulsar intensity changes. This approach not only reduces the influence of dynamic steady features but also further suppresses residual artefacts that persist after pre-processing (shown in Fig. 1) — including instrumental noise, radio frequency interference (RFI), and interstellar scattering.

To demonstrate the effectiveness of this approach, Fig. 6 shows that the pulsar will be effectively distinguished from steady features, regardless of whether it is surrounded by brighter or fainter steady features and noise.

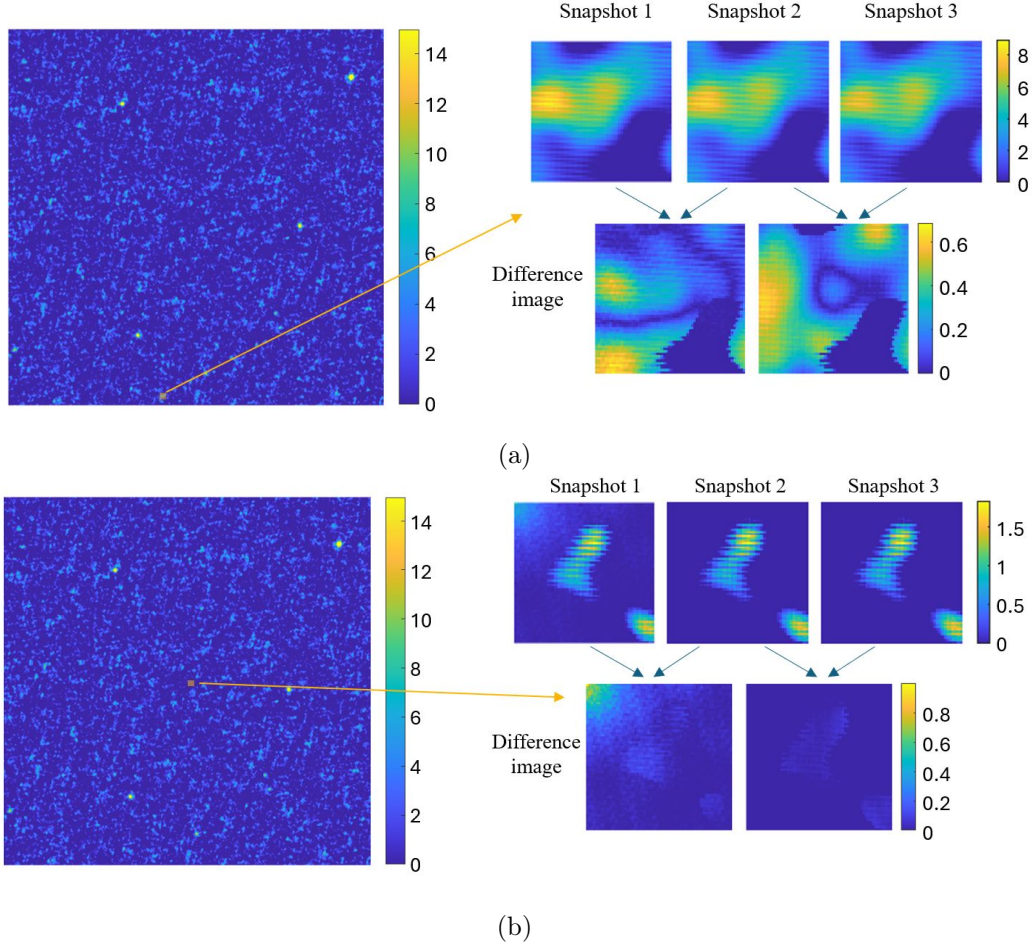


Figure 5: Example images of crops (a) and (b) from the simulated snapshots generated by OSKAR using SKA AA2 telescope layout. In (a), there is no pulsar, while in (b), a pulsar appears in the top-left corner of the tile. Snapshots 1, 2, and 3 are illustrated for both crops. The top row shows the snapshots, and the bottom row shows the difference images between adjacent snapshots.

2.2.4. Mathematics

Based on Sections 2.2.1 to 2.2.3, the equation of tLISI is given as

$$\begin{aligned}
 & \text{tLISI}_t^{(s)} \left(\mathbf{D}_t^{(s)}, \mathbf{D}_t^{(s+1)}, \mathbf{x}_t^{(s+1)} \right) \\
 &= 1 - \frac{\text{mean} \left(\left| \mathbf{D}_t^{(s)} - \mathbf{D}_t^{(s+1)} \right| \right) \max \left(\left| \mathbf{D}_t^{(s)} - \mathbf{D}_t^{(s+1)} \right| \right) \text{mean} \left(\mathbf{r}_t^{(s)} \right)}{\max \left(\max_{s=1}^L \mathbf{x}^{(s)}, C \right) \max \left(\max_{s=1}^L \mathbf{x}^{(s)}, C \right)}, \quad (2)
 \end{aligned}$$

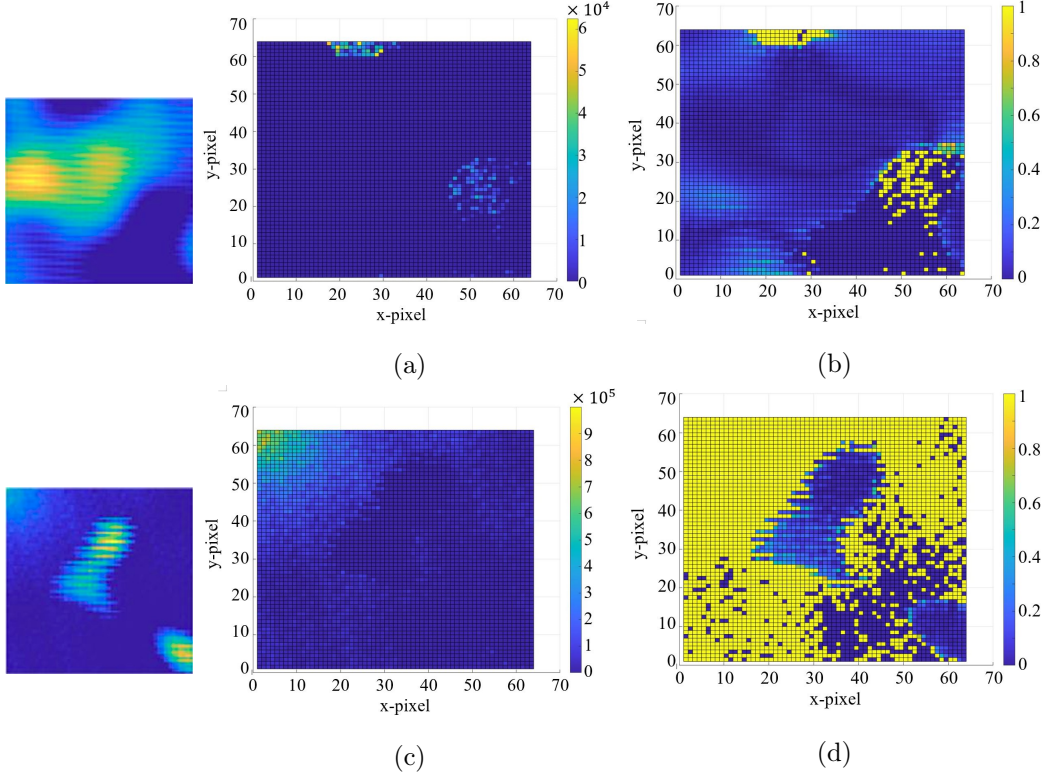


Figure 6: Effect of the \mathbf{r} -term. For clarity, the leftmost column reproduces the two crops of Snapshot 1 shown in Fig. 5 (a; top) and (b; bottom). Sub-figures (a) and (b) illustrate the \mathbf{r} values for the crop in Fig. 5 (a), before and after applying $r \rightarrow \min(r, 1)$, respectively, in the absence of a pulsar. On the other hand, sub-figures (c) and (d) demonstrate the corresponding \mathbf{r} values for the crop in Fig. 5 (b), which contains a pulsar, again before and after applying $r \rightarrow \min(r, 1)$. As shown in (d), the thresholded ratio maintains sensitivity to pulsar-induced variations (visible in the top-left corner) while reducing the response to steady features (central part).

where t is the tile index in an image, L is the total number of snapshots to be analysed, $s = 1, 2, \dots, L - 2$ is the snapshot index in the time sequential measurement, \mathbf{x} is the snapshot, \mathbf{D} is the difference image between adjacent snapshots which can be expressed by

$$\mathbf{D}_t^{(s)} = \left| \mathbf{x}_t^{(s+1)} - \mathbf{x}_t^{(s)} \right|, \quad (3)$$

\mathbf{r} is a ratio matrix, showing the ratio of intensity difference to its reference intensity level, which can be expressed by

$$r_{t,i}^s = \begin{cases} \frac{|D_{t,i}^{(s)} - D_{t,i}^{(s+1)}|}{\hat{x}_{t,i}^{(s+1)}} < 1, \\ 1, & \text{otherwise} \end{cases}, \quad (4)$$

where i is the pixel index in the tile and

$$\hat{x}_{t,i}^{(s+1)} = \begin{cases} C, & x_{t,i}^{(s+1)} = 0 \\ x_{t,i}^{(s+1)}, & \text{otherwise} \end{cases}. \quad (5)$$

C is adopted to avoid instability for a very small denominator, which is set to be 10^{-6} as it is smaller than the flux (in Jy) of most practical pulsars that have been observed (Hou et al., 2014; Wang et al., 2021). In this article, bold variables in mathematical equations denote vectors (arrays or matrices), unless otherwise specified.

To explain the tLISI equation in detail, we divide it into several parts.

- **Level of intensity difference**

The level of intensity gap between adjacent difference images is expressed as the normalised mean

$$\frac{\text{mean} \left(\left| \mathbf{D}_t^{(s)} - \mathbf{D}_t^{(s+1)} \right| \right)}{\max(\max_{s=1}^L \mathbf{x}^{(s)}, C)} = \frac{\Sigma \left| \frac{\mathbf{D}_t^{(s)}}{\max(\max_{s=1}^L \mathbf{x}^{(s)}, C)} - \frac{\mathbf{D}_t^{(s+1)}}{\max(\max_{s=1}^L \mathbf{x}^{(s)}, C)} \right|}{\Sigma 1}.$$

The numerator is divided by the maximum factor $\max(\max_{s=1}^L \mathbf{x}^{(s)}, C)$ to ensure that all images assessed by tLISI have comparable intensity levels. Using the maximum pixel value across all snapshots, rather than only the input pair of difference images, prevents artificially small denominators and resulting undesirably low tLISI values for similar low-intensity images. For real-time processing, this maximum can be estimated from at least three past snapshots. The maximum factor can also be considered as a normalisation factor, ensuring that tLISI remains between 0 and 1.

- **Representation of maximum change**

The maximum change between the two adjacent difference images is presented by

$$\frac{\max \left(\left| \mathbf{D}_t^{(s)} - \mathbf{D}_t^{(s+1)} \right| \right)}{\max \left(\max_{s=1}^L \mathbf{x}^{(s)}, C \right)}$$

to eliminate the influence of the intensity sums within the tiles (to address the case shown by Fig. 4).

- **Emphasis on pulsar change**

To determine the importance of variations, it is necessary to calculate the ratio of the difference $|D_{t,i}^{(s)} - D_{t,i}^{(s+1)}|$ to the intensity level $\hat{x}_{t,i}^{(s+1)}$, expressed as $\mathbf{r}_t^{(s)}$. A threshold of 1 is applied to each element $r_{t,i}^s$. The ratio can be interpreted as a weighting factor for the importance of changes.

The presence of the weighting factor explains why $\mathbf{x}_t^{(s+1)}$ is needed in the input of tLISI. When comparing the s -th difference image with the $(s + 1)$ -th difference image, the $(s + 1)$ -th snapshot is related to both difference images, thereby serving as the reference image. The $(s + 1)$ -th snapshot pixel-wise reflects the reference intensity level, regardless of whether both difference images include a pulsar, neither includes a pulsar, or only one includes a pulsar while the other does not (see Fig. 2).

The mean value of the weighting \mathbf{r} is used to measure the average level of importance of variations and to keep the range of tLISI between 0 and 1. The term $\mathbf{r}_t^{(s)}$ is crucial in tLISI for distinguishing pulsars from false positives, as shown in Fig. 6.

2.3. GPU-Accelerated tLISI

As Equation 2 shows, the calculation of each tile is independent in tLISI, allowing for parallel processing to improve computational efficiency.

Compute Unified Device Architecture (CUDA; Kirk 2007) is a parallel computing interface developed by NVIDIA Corporation. It enables users to directly programme NVIDIA GPUs, leveraging the parallel nature of GPUs through CUDA C programming. The CPU ⁴ and GPU-accelerated ⁵ tLISI

⁴<https://github.com/egbdfX/Intensity-sensitive-IQAs>

⁵<https://github.com/egbdfX/FastImagingTrigger>

are open source available on our Github. The tLISI-based FITrig is open-source available on Zenodo (Li, 2024a).

On the device, the most essential kernel is `tlisi`, whose pseudo code is shown in Algorithm 1. The target of this kernel is to calculate the tile-wised tLISI in an image with $N \times N$ pixels. According to Equation 2, tLISI is a function of $\Delta_t^{(s)} = \left| \mathbf{D}_t^{(s)} - \mathbf{D}_t^{(s+1)} \right|$. For the convenience of implementation, Equation 2 is transformed into

$$\mathbf{tLISI}^{(s)} \left(\Delta_t^{(s)}, \mathbf{x}_t^{(s+1)} \right) = 1 - \frac{\Sigma \Delta_t^{(s)} \max \Delta^{(s)} \Sigma \mathbf{r}_t^{(s)}}{(N_T \times N_T)^2 \times M^2}, \quad (6)$$

where $N_T \times N_T$ is the number of pixels in a tile and $M = \max \left(\max_{s=1}^L \mathbf{x}^{(s)}, C \right)$ is the maximum value of pixel intensities. The ‘‘mean’’ operation is transformed into a division of the sum, as the ‘‘max’’ and ‘‘sum’’ operations can be efficiently implemented using reduction operation in CUDA, leveraging shared memory in thread blocks. In this context, a thread block serves as a basic execution unit in parallel computing, where all threads inside the block are able to communicate via shared memory. Thereby, the reduction algorithm efficiently aggregates values from multiple threads by using shared memory to store intermediate results, recursively combining partial outputs in parallel until a final result is obtained. Here, the shared memory is used to compute three values: $\Sigma \Delta_t^{(s)}$, $\max \Delta^{(s)}$, and $\Sigma \mathbf{r}_t^{(s)}$.

To compute tiles in parallel, each tile is processed within a thread block, with the index of each tile denoted by (t_x, t_y) . This kernel consists of three segments.

- **Assignment:** Due to the independence between pixels in tLISI, each thread is responsible for the computations ($\Sigma \Delta_t^{(s)}$, $\max \Delta^{(s)}$, and $\Sigma \mathbf{r}_t^{(s)}$) for a set of pixels in the image, indexed by (p_x, p_y) :

$$\begin{cases} p_x = t_x \times N_T + \lfloor \frac{\mathbf{threadIdx.x} + (k-1) \times \mathbf{blockDim.x}}{N_T} \rfloor \\ p_y = t_y \times N_T + \text{mod}(\mathbf{threadIdx.x} + (k-1) \times \mathbf{blockDim.x}, N_T) \end{cases}, \quad (7)$$

where $k = 1, 2, \dots, K$ with $K = \lceil N_T \times N_T / \mathbf{blockDim.x} \rceil$ such that $\mathbf{threadIdx.x} + (k-1) \times \mathbf{blockDim.x} < N_T \times N_T$. The corresponding pseudo code is shown in Segment 1 of Algorithm 1.

- **Reduction:** The sums and maximum value of all threads within a block are calculated by the reduction operation, as shown in Segment 2 of Algorithm 1.

Algorithm 1: Pseudo code (GPU kernel) for the shared-memory-based tLISI

Input: $\Delta^{(s)}$, $\mathbf{x}^{(s+1)}$, N , N_T , M

Output: tLISI

Kernel tlisi($\Delta^{(s)}$, $\mathbf{x}^{(s+1)}$, N , N_T , M):

```

// Shared memory buffer
__shared__ B1,2,3 // B1, B2, B3 for  $\Sigma\Delta$ ,  $\max\Delta$ ,  $\Sigma r$ 
// Indices of the tile
 $t_x \leftarrow \lfloor \text{blockIdx.x} / (N/N_T) \rfloor$ ,  $t_y \leftarrow \text{mod}(\text{blockIdx.x}, N/N_T)$ 
 $K \leftarrow \lceil N_T \times N_T / \text{blockDim.x} \rceil$ 

// Segment 1: Assignment
for  $k \leftarrow 1$  to  $K$  do
    if  $\text{threadIdx.x} + (k - 1) \times \text{blockDim.x} < N_T \times N_T$  then
        if  $k == 1$  then
             $B_1[\text{threadIdx.x}] = 0$ 
             $B_2[\text{threadIdx.x}] = 0$ 
             $B_3[\text{threadIdx.x}] = 0$ 

            // Indices of pixels
             $p_x \leftarrow t_x \times N_T + \lfloor (\text{threadIdx.x} + (k - 1) \times \text{blockDim.x}) / N_T \rfloor$ 
             $p_y \leftarrow$ 
                 $t_y \times N_T + \text{mod}(\text{threadIdx.x} + (k - 1) \times \text{blockDim.x}, N_T)$ 

            // Assignment in shared memory
             $B_1[\text{threadIdx.x}] += \Delta^{(s)}[p_x \times N + p_y]$ 
             $B_2[\text{threadIdx.x}] = \max$ 
                 $(B_2[\text{threadIdx.x}], \Delta^{(s)}[p_x \times N + p_y])$ 
            if  $\Delta^{(s)}[p_x \times N + p_y] / \mathbf{x}^{(s+1)}[p_x \times N + p_y] < 1$  then
                 $B_3[\text{threadIdx.x}] +=$ 
                     $\Delta^{(s)}[p_x \times N + p_y] / \mathbf{x}^{(s+1)}[p_x \times N + p_y]$ 
            else
                 $B_3[\text{threadIdx.x}] += 1$ 
        else
            if  $k == 1$  then
                 $B_1[\text{threadIdx.x}] = 0$ 
                 $B_2[\text{threadIdx.x}] = 0$ 
                 $B_3[\text{threadIdx.x}] = 0$ 

```

Algorithm 2: Pseudo code (GPU kernel) for the shared-memory-based tLISI (continue)

Input: $\Delta^{(s)}$, $\mathbf{x}^{(s+1)}$, N , N_T , M

Output: tLISI

Kernel tlisi($\Delta^{(s)}$, $\mathbf{x}^{(s+1)}$, N , N_T , M):

```

// Segment 2: Reduction
for d ← blockDim.x/2 to 0 by d = d/2 do
    // Synchronize threads
    __syncthreads()
    if threadIdx.x < d then
        B1[threadIdx.x] += B1[threadIdx.x + d]
        B2[threadIdx.x] = max (B2[threadIdx.x],
                               B2[threadIdx.x + d])
        B3[threadIdx.x] += B3[threadIdx.x + d]

// Segment 3: Output
if threadIdx.x == 0 then
    tLISI[blockIdx.x] = 1 -
        (B1[0]/NT/NT) × B2[0] × (B3[0]/NT/NT)/M/M

```

- **Output:** In Segment 3 of Algorithm 1, the tLISI value of tile (t_x, t_y) (expressed by `blockIdx.x`) is calculated and stored.

The computational complexity can be analysed by examining each segment. In the assignment stage, each thread independently computes intermediate quantities for a subset of pixels within a tile of size $N_T \times N_T$. Serially, this requires $O(N_T^2)$ operations per tile; however, the independence of pixels allows full parallelisation across threads within a block, reducing the effective per-tile runtime roughly to $O(N_T^2/n_{\text{threads}})$, where n_{threads} is the number of threads per block. Across all tiles in an image of size $N \times N$, the total number of operations in the assignment stage scales as $O(N^2)$. Because of the computations for pixels and tiles are independent, GPU parallelisation allows threads and tiles to be processed concurrently, significantly reducing the wall-clock runtime compared to serial execution.

In the reduction stage, sums and maxima are aggregated across threads using shared-memory reduction. This operation has a logarithmic complexity

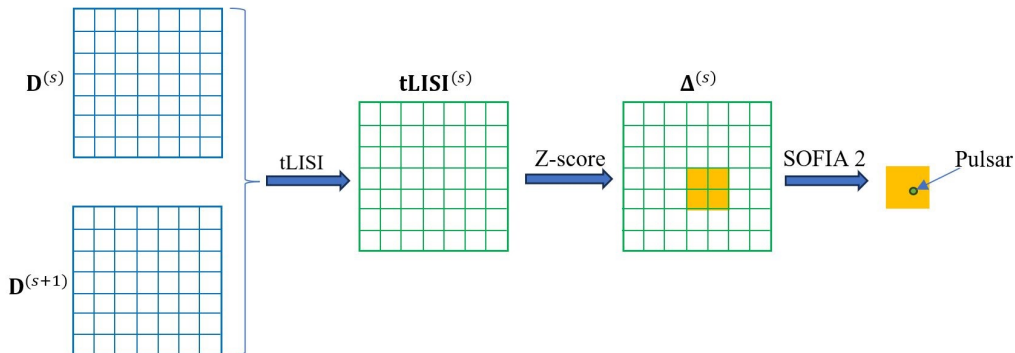


Figure 7: Diagram of FITrig: image domain detection approach. Here, the current state-of-the-art source finder, SOFIA 2, is used as an example to demonstrate the functionality of FITrig, though it could be replaced by other source finders.

$O(\log N_T^2)$ per block, which is minor compared to the assignment stage. Finally, the output stage performs a constant-time computation per tile ($O(1)$).

The GPU’s highly parallel architecture substantially reduces the effective runtime, making tLISI suitable for rapid processing of large images.

2.4. Fast Imaging Trigger (FITrig)

FITrig consists of two components: localising potential regions (using tLISI) and pulsar searching (in the image or image-frequency domain). The technique is named “Trig” because it operates as a trigger-based detection technique, where tLISI serves as the core mechanism for identifying pulsar signals. FITrig is developed in two complementary versions for pulsar searching — the fundamental image domain FITrig (Section 2.4.1) and its enhanced image-frequency domain FITrig (Section 2.4.2).

2.4.1. Image Domain Detection

The image domain detection approach of FITrig is illustrated in Figure 7.

The tLISI algorithm outputs a $\frac{N}{N_T} \times \frac{N}{N_T}$ tLISI matrix $\mathbf{tLISI}^{(s)}$ that indicates the similarity between the difference images $\mathbf{D}^{(s)}$ and $\mathbf{D}^{(s+1)}$. Each element of the tLISI matrix denotes the tLISI value of the corresponding tile in the difference images. Subsequently, to statistically identify candidate tiles likely to contain pulsar(s), the z-score is calculated for the tLISI matrix,

as shown in

$$z_t^{(s)} = \frac{(1 - \text{tLISI}_t^{(s)}) - (1 - \mu^{(s)})}{\sigma^{(s)}} = \frac{\mu^{(s)} - \text{tLISI}_t^{(s)}}{\sigma^{(s)}}, \quad (8)$$

where t is the tile (element) index in the tLISI matrix and z-score matrix, z_t is the z-score for the t -th tile, tLISI_t is the tLISI value for the t -th tile, and μ and σ represent the mean and standard deviation of the tLISI matrix, respectively. Note that the complement of tLISI is used in the calculation of the z-score because the greater the difference between the two images, the closer the tLISI value is to 0. Therefore, the similarity of the two images is represented by the value of tLISI, while the difference is represented by the value of the complement of tLISI.

The tiles of the image with z-scores higher than a threshold τ (discussed in Section 3) will be selected as candidate tiles. Typically, a pulsar is often included in more than one tile, especially when the tile size is very small. In such cases, the tLISI step will forward all relevant tiles to the source finder, e.g., SOFIA 2, for accurate localisation of the pulsar. It will output the positions of pulsars in Right Ascension and Declination (RA/Dec), along with their fluxes. The set of indices for the selected tiles is expressed as

$$\left\{ t | z_t^{(s)} > \tau \right\}. \quad (9)$$

2.4.2. Image-Frequency Domain Detection

As analysed in Section 2.1, ULP pulsars exhibit sparse temporal distributions. For detecting these pulsars in datasets containing unknown sources, the image-frequency domain FITrig proposed in this section provides a more robust localisation solution, building upon the fundamental image domain FITrig (Section 2.4.1).

Based on the periodicity of ULP pulsars, Fig. 8 shows the image-frequency domain detection approach of FITrig.

In this approach, the tLISI algorithm is applied to time-sequential difference images \mathbf{D} to obtain a $\frac{N}{N_T} \times \frac{N}{N_T} \times (L - 2)$ tLISI cube, constructed from tLISI values for tiles in each pair of adjacent difference images, with the third dimension representing time.

FITrig executes a fast Fourier transform (FFT) on the sequential tLISI values corresponding to each tile over the time axis. This process results in a 3-dimensional (3-D) cube, where the first two axes represent the positions of tiles and the third dimension represents frequency. Subsequently,

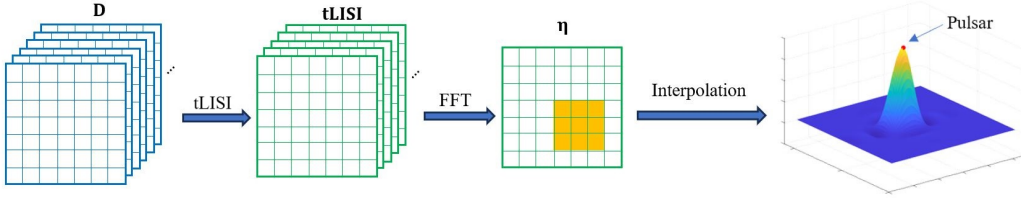


Figure 8: Diagram of FITrig: image-frequency domain detection approach.

a spectrum matrix η is constructed, with each element η_t representing the informative magnitude of the frequency spectrum along the third axis for the corresponding tile t , as shown in

$$\eta_t = h(|\mathcal{F}[\text{tLISI}_t(s) - \text{mean}(\text{tLISI}_t(s))]|), \quad (10)$$

where $\mathcal{F}[\bullet(s)]$ indicates the Fourier transform operation over the s -axis and h indicates a spectral refinement function, such as harmonic summing (Adámek et al., 2022), to form informative spectrum. The mean of the tLISI_t sequence, represented by the Direct Current (DC) component from the spectrum, is removed.

The values in the spectrum matrix η act as contributions of the pulsar(s) to different tiles. Typically, a pulsar primarily contributes to either one tile, a cluster of two tiles (when the pulsar is on the boundaries of tiles), or a cluster of four tiles (when the pulsar is at the corners of tiles). All potential tiles that may contain pulsar(s) are selected by applying a threshold τ (discussed in Section 3) on the z-score matrix of the spectrum matrix, as expressed by

$$z_t = \frac{\eta_t - \text{mean}(\eta)}{\text{std}(\eta)}. \quad (11)$$

The set of indices for the selected tiles is given by

$$\{t | z_t > \tau\}. \quad (12)$$

To reconstruct the pulsar position based on its contributions across different tiles, interpolation methods are activated for precise pulsar localisation in the selected regions. Unlike pixel-level methods such as 2-D Gaussian fitting that operate on snapshots, our tile-based approach using bilinear, spline, bicubic, or Makima cubic interpolation offers computational advantages while maintaining precision. This method is effective for real-time automation of localising ULP pulsars in large-area surveys.

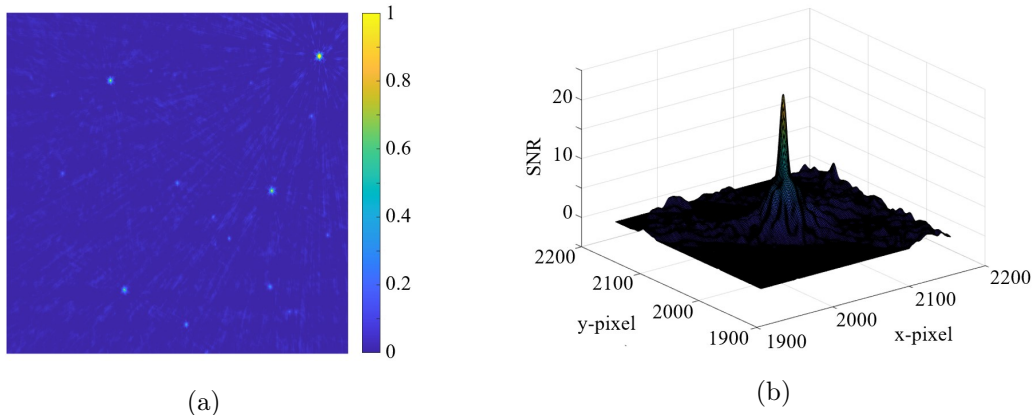


Figure 9: FOV centred on PSR J0901-4046, observed with MeerKAT, with an image size of 4096×4096 pixels. Intensities are normalised for illustration purposes. (a) Dirty snapshot in which the pulsar exhibits its highest intensity within the dataset. (b) SNR plot corresponding to (a), showing only the regions around the pulsar.

The image-frequency domain FITrig takes into account the time-sequential tLISI, which further reduces the influence of noise.

3. Performance

We evaluate FITrig in two main settings. The first involves testing the real-world measurement of PSR J0901-4046 to examine its performance under practical conditions and realistic data variability. The second comprises a series of experiments on simulated datasets, providing controlled settings to evaluate FITrig’s robustness across diverse observational conditions.

Real Dataset Evaluation

We evaluate FITrig on a real dataset of PSR J0901-4046, a periodic pulsating neutron star with a period of 76 seconds, observed by MeerKAT in the L band (Caleb et al., 2022). The Measurement Set contains 1500 time samples of visibilities, with a sampling period of 2 seconds. Figure 9 shows a 1.2-degree Field of View (FOV) centred on the pulsar in sub-figure (a), along with the corresponding SNR plot in sub-figure (b), where the pulsar reaches an SNR of 22.7.

To search for the ULP pulsar in the dataset, we use FITrig, whose processing steps are illustrated in Fig. 8, with harmonic summing applied to the

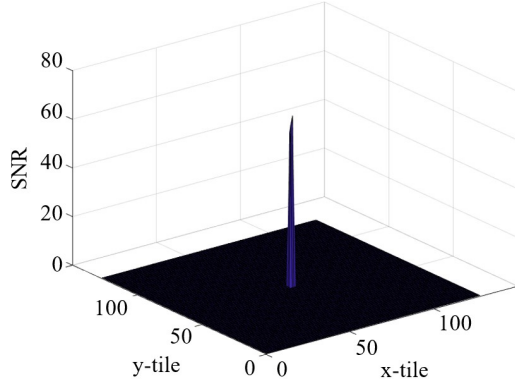


Figure 10: SNR plot of spectrum matrix η obtained by applying FITrig ($N_T = 32$) to the real dataset of PSR J0901-4046.

spectrum. Figure 10 shows the SNR plot of the spectrum matrix η , in which the pulsar appears as a distinct peak with an SNR of 70.3, higher than the SNR in any snapshot.

We apply spline interpolation to the spectrum matrix to strike a balance between precision and computational efficiency (Li, 2024b). The interpolation ensures that the peak position, corresponding to the pulsar position, aligns with its actual coordinates at RA 135.37° , Dec -40.77° , yielding arcsecond-level localisation accuracy. After detecting the ULP pulsar from the dataset with FITrig, higher localisation precision can also be achieved by activating a source finder, such as SOFIA 2, on a tile containing the pulsar. The resulting localisation accuracy then depends on the precision of the source finder.

Focusing on the pulsar tile, the harmonic summed spectrum (Fig. 11) clearly reveals a frequency of 0.013 Hz, corresponding to a 76-second period. This demonstrates that FITrig not only localises the pulsar with high precision but also recovers its accurate spin frequency, further confirming the effectiveness of FITrig.

In addition to its accuracy, FITrig offers a substantial advantage in computational efficiency. In the experiments, FITrig is executed on the NVIDIA H100 ⁶. With high levels of parallelisation and GPU acceleration, FITrig can complete pulsar detection and localisation in 6.34 seconds on this 1500-

⁶<https://resources.nvidia.com/en-us-tensor-core>

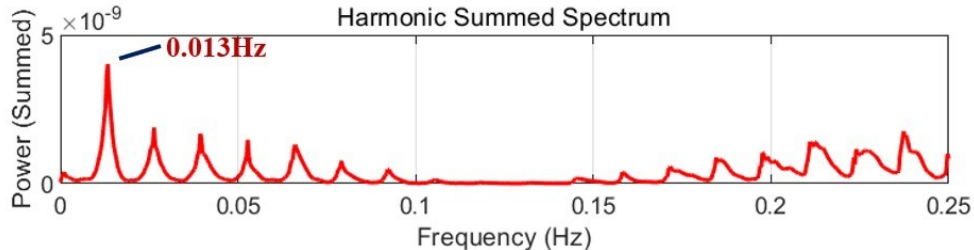


Figure 11: Harmonic summed spectrum corresponding to the time-sequential images of the pulsar tile. The fundamental frequency in the spectrum is 0.013 Hz, corresponding to the spin frequency of PSR J0901-4046.

snapshot dataset, covering the process from image loading to pulsar position output. This performance is superior to that of existing pulsar detection techniques and can be regarded as real-time. For details of the computational efficiency of FITrig, a more comprehensive discussion is provided in Section 3.2. This includes consideration of very large-scale images (e.g., $50\text{K} \times 50\text{K}$ pixels), as next-generation telescopes can produce data at this scale, where existing methods are computationally challenging for pulsar searches on such large images.

As shown by the real-data evaluation, FITrig demonstrates accurate localisation of the ULP pulsar and precise frequency detection at very high speed. In fact, beyond computational efficiency, it also achieves a higher sensitivity (Section 3.3) and a lower false positive rate (Section 3.4) than existing methods, which will be further evaluated on simulated datasets under a variety of conditions in the following section.

Simulation-Based Evaluation

3.1. Parameters and Experimental Settings

We evaluate FITrig using simulated datasets across various settings and parameter configurations. To avoid bias from different telescope layouts, we simulate images for the same SBD using telescope configurations of MeerKAT and SKA AA2. The sampling rate is tested across a range from 0.016 seconds (or smaller) to 16 seconds, where the lower bound exceeds the capabilities of current telescopes as our design targets next-generation telescopes or beyond, while the upper bound exceeds the sampling rates achievable by current techniques, demonstrating the practicality of our method for existing telescopes.

To evaluate FITrig performance under varying conditions, the three-snapshot unit (as shown in Fig. 3) is created following three key principles.

Firstly, images are based on the real Measurement Set to represent practical scenarios. Specifically, we image and deconvolve snapshots from the real Measurement Set, then use the resulting restored image as a sky model in OSKAR to produce simulated Measurement Sets using MeerKAT and SKA AA2 telescope layouts, separately. Secondly, images are designed to simulate the behaviour of ULP pulsars during observations. To be more precise, in the three-snapshot unit, the pulsar is only visible in one of the three snapshots. W-Stacking Clean (WSClean; Offringa et al. 2014) is used to reconstruct dirty snapshots from Measurement Sets. Thirdly, several sets of images are produced with steady features at different intensity levels. More concretely, in the simulation, the relative brightness between steady features and the pulsar is controlled by $\Gamma = 0.05, 1, 20, 40, 60, 80, 100$, defined as the ratio of the brightest steady features to the pulsar flux.

Three important parameters (threshold τ , tile size $N_T \times N_T$, length of time-sequential snapshots L) are explained below. To begin with, the parameter τ serves as the threshold for z-score values, filtering significant cases by indicating how many standard deviations (σ) the tLISI of pulsar deviates from the average level (μ). We investigate sigma levels ranging from 3 to 6, with 5σ as the reference level as it is commonly used in transient detection studies (Bhardwaj et al., 2021; Hancock et al., 2012).

With regard to the second parameter, tile size, we use the number of pixels to represent effective tile size N_T , rather than the degrees of FOV covered by a tile, to avoid the influence of different telescope resolutions. Table 2 illustrates the effective tile sizes for different combinations of threshold τ , telescope arrays, and relative brightness Γ . The “-” entry indicates that, for the given parameters, no valid effective tile size can be determined. Here, the phrase “effective tile sizes” refers to tile sizes for which the pulsar tile is successfully selected while keeping the proportion of selected pixels below a level of $\theta = 0.5\%$ (Li, 2024b). This level θ controls the trade-off between false positives and detection sensitivity — lower θ reduces false positives but also decreases sensitivity. For example, when $\theta = 0.1\%$ and steady sources are very bright ($\Gamma > 20$), the pulsar becomes undetectable in our experiments.

In the results, the pulsar detected by the FITrig can be very faint, with its flux reaching between 1/40 and 1/60 of that of the surrounding steady features in the images using MeerKAT and SKA AA2, respectively, for $\tau = 5$. A very small tile size (e.g., per-pixel detection) amplifies the impact of SNR fluctuations. Comprehensively, the best tile size for pulsar localisation is 32×32 pixels. While we adopt this size, FITrig supports customisable tile sizes

| Effective N_T | | Γ | | | | | | |
|-----------------|----------|----------|-------|-------|-------|------|------|------|
| Telescope | τ | 0.05 | 1 | 20 | 40 | 60 | 80 | 100 |
| MeerKAT | 3 | 1-128 | 1-128 | 1-64 | - | - | - | - |
| | 4 | 1-128 | 1-128 | 1-64 | 32-64 | - | - | - |
| | 5 | 1-128 | 1-128 | 1-64 | 8-64 | - | - | - |
| | 6 | 1-128 | 1-128 | 1-256 | 1-128 | 1-64 | 1-64 | 1-64 |
| SKA AA2 | 3 | 1-128 | 1-64 | 2-64 | - | - | - | - |
| | 4 | 1-128 | 1-128 | 1-128 | 16-32 | - | - | - |
| | 5 | 1-128 | 1-128 | 1-128 | 1-64 | 32 | - | - |
| | 6 | 1-128 | 1-128 | 1-256 | 1-128 | 1-64 | 1-64 | 1-32 |

Table 2: Effective tile sizes under different choices of circumstances (telescope layouts, significance threshold τ , pulsar faintness Γ), with the number of potential false positives to be less than $\theta = 0.5\%$ number of pixels of the whole image.

to accommodate diverse user needs.

Regarding the third parameter L , longer observation durations (larger L at constant sampling period T_s) enhance signal detection through greater energy accumulation and reduced noise, but require processing more snapshots, creating a real-time performance trade-off.

In the experiments, we simulate three pulsars under different sampling conditions: one undersampled, one nearly critically sampled, one oversampled, according to the Nyquist criterion. In the experiment, various lengths of snapshots are performed, including $L = 258, 514, 1026$, and 2050 , ensuring that the number of sequential tLISI matrices over time ($L - 2$) is a power of 2 for computational purposes. The pulsar profile is simulated as a truncated Gaussian (Lorimer and Kramer, 2004) to avoid power leakage between pulsar periods. The duty cycle of the pulsar is set to 10% of the pulsar period. Within each sampling period, the pulsar intensities are accumulated and integrated to obtain flux.

Please note that in this article we demonstrate FITrig on typical on-and-off pulsars; a more comprehensive evaluation of its generalisation, including cases with multiple transients in the same FOV and pulsars exhibiting gradual variations in brightness over time, will be presented in an upcoming paper on the Fast Imaging Pipeline in this series.

3.2. Computational Efficiency

In FITrig, each tile is computed in parallel, as explained in Section 2.3. In image domain FITrig, SOFIA 2 only needs to process a few tiles rather than the whole image, which effectively speeds up the computing process. The enhancement is more evident when the image is larger. To evaluate computational performance, we input images filled with random Gaussian noise. This approach ensures consistency, because the computation time of FITrig depends solely on pixel intensities and remains unaffected by image content. In contrast, SOFIA 2’s runtime varies depending on the number and complexity of sources it detects, since it performs feature extraction. Using these inputs, we eliminate content-related variables and focus solely on comparing the underlying computational efficiency of both methods.

In the experiments, images with sizes ranging from 512×512 pixels to $50K \times 50K$ pixels are inputted separately. For each image size, 10 tests are run and the median computing time among the 10 tests is calculated. Three consecutive snapshots are inputted for each run. Since the tiles selected by the tLISI algorithm will be forwarded for further localisation of the pulsars, SOFIA 2 will run on the selected tiles.

The results of the computation time are shown in Fig. 12. The computation time of the tLISI algorithm, excluding I/O time between host and device as well as memory allocation time, is also shown in the results (as “excl. I/O”). This is because when the tLISI is applied in the middle of a FI pipeline, the data is already available on the device, so there is no need to carry out memory allocation and data transfer between the host and the device.

According to the results, when the image size increases, our FITrig speeds up processing, which can be 4.3 times faster than solely applying SOFIA 2 when the image size reaches $50K \times 50K$ pixels.

In image-frequency domain FITrig, a set of time-sequential snapshots needs to be loaded onto the device. Rather than loading all snapshots at once, a better way to compute is to cover the data transfer time on Peripheral Component Interconnect Express (PCIe) as much as possible by the computing process of tLISI. CUDA kernels running on one group of inputs (three consecutive snapshots) are called “trigger kernel”. The output of each trigger kernel is a tLISI matrix.

To design the computing process, we measure the time of transferring a snapshot (input) and a tLISI matrix (output) over PCIe Gen 4.0 using

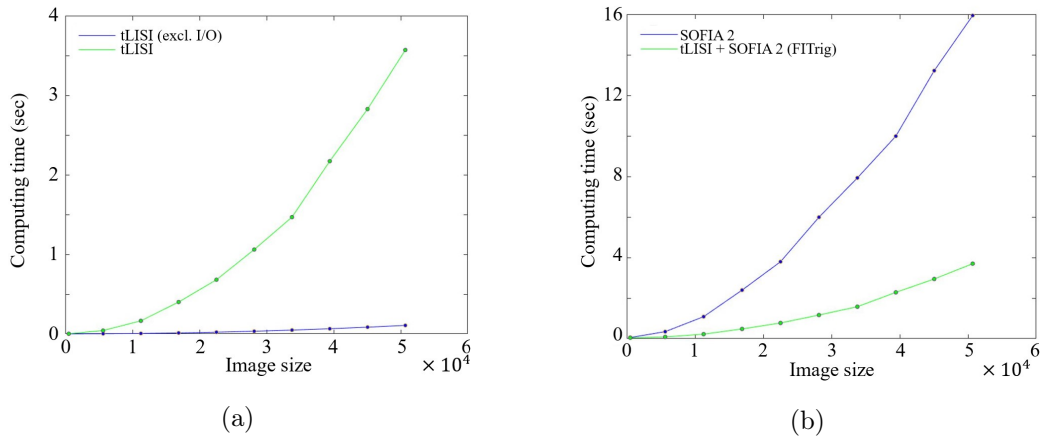


Figure 12: Computation time, where (a) compares the computation time excluding (blue lines) and including (green lines) I/O time between host & device and memory allocation time, and (b) compares the computation time for FITrig (green lines) and using SOFIA 2 alone (blue lines). The image size is represented by N .

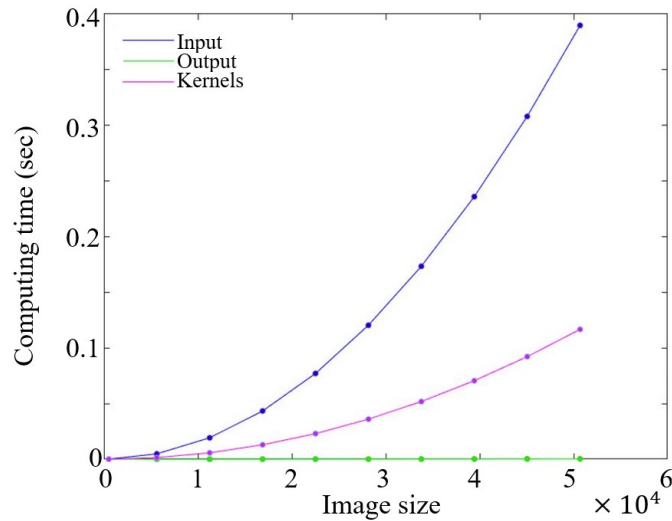


Figure 13: Time of data transfer (input and output) with pinned memory vs. time of the trigger kernel against image size (N).

pinned memory. The time of data transfer and the time of trigger kernel for different sizes of images are shown in Fig. 13. Assuming the same utilisation, switching from PCIe 4.0 to PCIe 5.0 will double the speed due to the increased bandwidth.

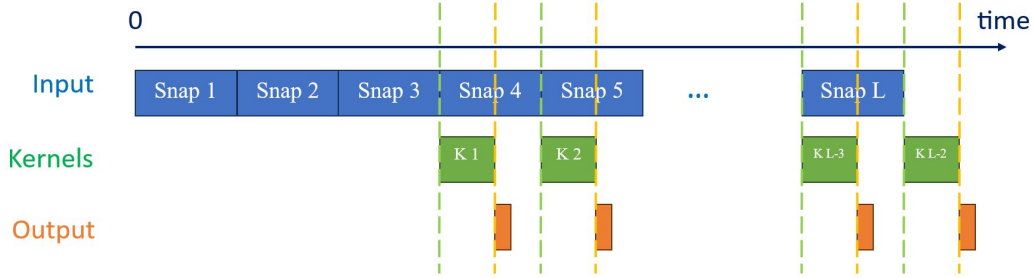


Figure 14: Parallel timeline of the trigger: “Snap s ” ($s = 1, 2, \dots, L$) indicates the s -th snapshot, “K n ” ($n = 1, 2, \dots, L - 2$) indicates the n -th trigger kernel process, and the orange bar (too small for a label) indicates the n -th ($n = 1, 2, \dots, L - 2$) output tLISI matrix. “Input” indicates the process of transferring a snapshot from the host to the device, “Kernels” indicates the execution of trigger kernels on the device, and “Output” indicates the process of transferring a tLISI matrix from the device to the host.

As shown in Fig. 13, the time required to output a tLISI matrix is very short, and can be ignored compared to the kernel processing time and the input transfer time. To improve efficiency, the design of the parallel timeline is shown in Fig. 14. Specifically, three consecutive snapshots are needed for each tLISI computation. At the beginning, the first three snapshots are loaded onto the device. Then, while the n -th ($n = 1, 2, \dots, L - 3$) tLISI is being computed, the $s = n + 3$ th snapshot is loaded. The $n + 1$ th tLISI computation begins once the $s = n + 3$ th snapshot has finished loading. The $n = L - 2$ th tLISI will be the last, during which no further data is loaded. The n -th ($n = 1, 2, \dots, L - 2$) tLISI result is transferred once the n -th tLISI process is done.

3.3. Sensitivity

Current telescopes like MeerKAT can achieve sampling periods of 2 or 8 seconds, while next-generation telescopes such as the SKA will reach 1-second sampling period (Dewdney et al., 2016). Future telescopes may further reduce this duration. Detection sensitivity is related to the sampling period T_s as it affects signal integration within a snapshot. Sections 3.3.1 and 3.3.2 analyse detection strategies and sensitivity for long and short sampling periods — where “long” and “short” are relative descriptions. The goal is to determine FITrig’s maximum tolerable sampling period and to explore mitigation strategies for scenarios where future telescopes employ extremely short sampling periods. Section 3.3.3 presents the detection sensitivity of FITrig

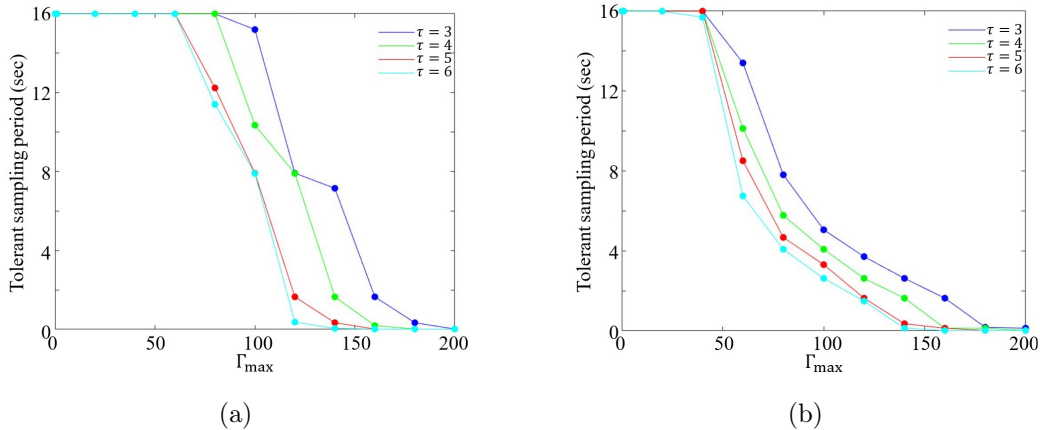


Figure 15: Allowable sampling period under various settings of Γ and τ using telescope array (a) MeerKAT and (b) SKA AA2.

across varying sampling periods corresponding to different Nyquist sampling conditions.

3.3.1. Detection Sensitivity Analysis for Long Sampling Periods

Larger T_s values lead to increased variations in steady features of consecutive snapshots. Therefore, it is crucial to determine the maximum allowable T_s when applying the FITrig. As shown in Fig. 15, experiments reveal the maximum allowable T_s for various Γ values using configurations of MeerKAT and SKA AA2. The allowable sampling period is obtained by comparing the z-score of pulsar tile to a threshold τ . If the z-score surpasses τ (i.e., the pulsar is detected), the corresponding sampling period is considered “allowable”; otherwise, it is not. In the experiments, the maximum sampling period evaluated is 16 seconds, which is sufficient to show the cases for pulsar detection.

The findings indicate that the appropriate sampling period for applying FITrig in detection is influenced by both the telescope configuration and the desired sensitivity. The allowable sampling period is constrained by the intensity of steady features in the same SBD. For instance, detecting a pulsar 100 times fainter than the brightest surrounding steady features, and using a threshold of 6, the sampling period must not exceed 7.9 seconds for MeerKAT and 2.6 seconds for SKA AA2.

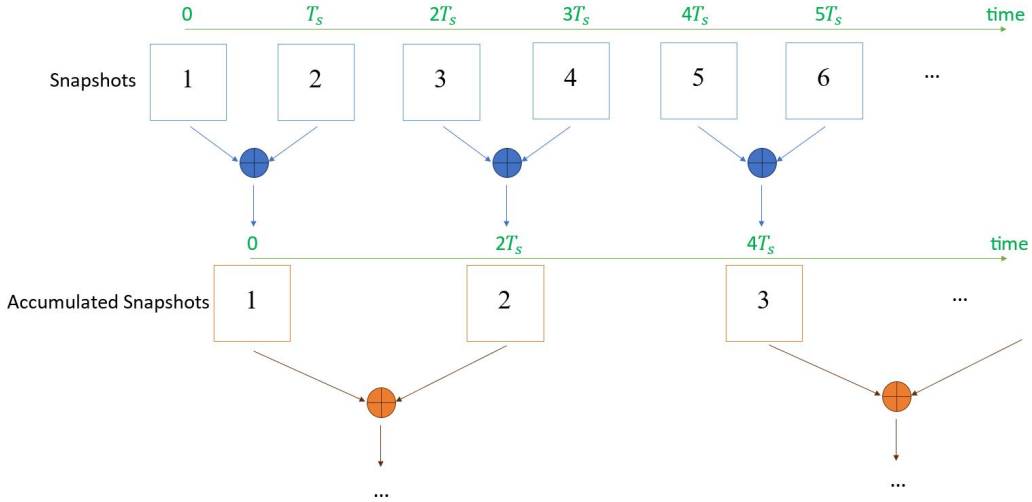


Figure 16: Snapshots accumulation, where the symbol of “ \oplus ” means adding the two snapshots together and outputting their sum.

3.3.2. Sensitivity Limit and Mitigation Approach for Short Sampling Periods

An extremely short T_s — especially with advanced telescopes that offer ultra-high sampling rates — results in consecutive snapshots being almost identical, whether they capture transient or steady features. In such scenarios, applying FITrig to near-blank difference images is impractical.

To address this, when adjacent snapshots are very similar, FITrig can be applied after combining multiple snapshots, as illustrated in Fig. 16. This approach ensures that the inputs fed into FITrig show noticeable variations. Snapshots can be combined progressively until there is enough difference between adjacent accumulated snapshots. The accumulation process preserves all information while increasing the contrast between the snapshots.

3.3.3. Nyquist Sampling Effects on Detection Sensitivity

With increasing Γ , the pulsar becomes relatively fainter and is more easily buried in noise. The sensitivity of the image-frequency domain FITrig is evaluated by the z-score of the pulsar tile, as a measure of the SNR. Tiles with z-scores above the threshold $\tau = 6$ are forwarded for localisation. In this test, we employ the maximum spectral magnitude rather than harmonic summing to assess the detection robustness under conservative conditions, ensuring that our method remains effective even when harmonic power is negligible or obscured by noise.

Z-scores of the pulsar tile under different circumstances (pulsar frequency f_p , telescope arrays, L and Γ) are shown in Table A.6. In it, f_s is the sampling rate of the telescope array, and ‘-’ means that the z-score of the pulsar tile under a certain combination of those circumstances is lower than $\tau = 6$, indicating that the pulsar cannot be detected. According to the Nyquist-Shannon sampling theorem, when $f_s/(2 \times f_p) \geq 1$, it is proper sampling; otherwise, it is under sampling.

From the table, as Γ increases, the z-score of the pulsar tile decreases for the same L , telescope array, and f_p . This suggests that the pulsar becomes relatively fainter, making it harder to detect. When the length of the snapshot series (L) is longer, the η_t of the pulsar tile becomes larger. However, L does not have a significant effect on z_t , because the detection probability remains stable while longer series produce stronger spectral amplitudes. Thus, the detection will not be affected by L , as long as it is not too short.

Furthermore, when $f_s/(2 \times f_p)$ is larger, the pulsar being detected can have a larger Γ , indicating that the pulsar is relatively fainter. Even if $f_s/(2 \times f_p)$ is slightly smaller than 1, a pulsar with flux 20 times fainter than the surrounding steady features can be detected. In contrast, when $f_s/(2 \times f_p)$ is much smaller than 1, where the pulsar signal is under sampled, only bright pulsars can be detected.

3.4. False Positive Rate

3.4.1. Image Domain FITrig

By applying tLISI before the source finder SOFIA 2, not only is the computation sped up, but the false positive rate is also reduced. Using the simulated data, Table 3 shows the number of candidates detected by SOFIA 2 with and without tLISI under different Γ . For higher precision, τ is set to 6. The term “# Candidates” indicates the number of output sources from SOFIA 2. A single pulsar is placed in the images. Two approaches are compared: “FITrig” refers to using tLISI to activate SOFIA 2 on relevant tiles, while “SOFIA2” refers to applying SOFIA 2 on the whole image.

Currently, astronomers apply clustering algorithms to the candidates outputted by SOFIA 2 to determine whether they are true pulsars. As shown in the table, the number of candidate transients is greatly reduced (by a factor of up to 858.8) when using FITrig, making it easier for users to select the true pulsar(s) among the candidates.

| # Candidates | | Γ | | | | | | |
|--------------|----------|----------|------|------|------|------|------|------|
| Telescope | Approach | 0.05 | 1 | 20 | 40 | 60 | 80 | 100 |
| MeerKAT | FITrig | 8 | 8 | 9 | 38 | 67 | 90 | 93 |
| | SOFIA2 | 3989 | 3970 | 3868 | 3851 | 3846 | 3844 | 3841 |
| SKA AA2 | FITrig | 4 | 4 | 5 | 30 | 62 | 72 | 75 |
| | SOFIA2 | 3435 | 3428 | 3346 | 3339 | 3331 | 3329 | 3327 |

Table 3: Number of candidate transients in image domain detections.

| # Candidates | | Γ | | | | | | |
|--------------|----------------------|----------|---|----|----|----|----|-----|
| Telescope | $f_s/(2 \times f_p)$ | 0.05 | 1 | 20 | 40 | 60 | 80 | 100 |
| MeerKAT | 0.0982 | 1 | 1 | - | - | - | - | - |
| | 0.9817 | 1 | 1 | 38 | - | - | - | - |
| | 9.8175 | 1 | 1 | 14 | 34 | 38 | - | - |
| SKA AA2 | 0.0982 | 1 | 1 | - | - | - | - | - |
| | 0.9817 | 1 | 1 | 28 | - | - | - | - |
| | 9.8175 | 1 | 1 | 1 | 16 | 25 | 31 | 32 |

Table 4: Number of candidate transients localised by image-frequency domain FITrig.

3.4.2. Image-Frequency Domain FITrig

By image-frequency domain FITrig, the number of candidate pulsars after the interpolation is further reduced. With the threshold of $\tau = 6$ and the sequence length of the snapshots of $L = 1026$ (a reasonable choice to maintain both sensitivity and computational efficiency), the number of candidate transients corresponding to different circumstances is shown in Table 4. Compared to the current source finder SOFIA 2, as shown in Table 3, the number of false positives is greatly reduced.

3.4.3. False Positives for Non-Pulsar Snapshots

Different sampling rates impact the number of false positives when images in a three-snapshot unit are all non-pulsar. The amount of false positives is influenced by the dirty beam associated with each telescope configuration.

The selection of pulsar candidates is based on statistical analysis using z-scores. As such, calculating z-scores solely on the tLISI matrix obtained from non-pulsar snapshots is not sensible. By calculating z-scores for tLISI of each tile, incorporating tiles from both pulsar and non-pulsar snapshots, we identify false positives in the non-pulsar snapshots. This is achieved by

activating SOFIA 2 on tiles with z-score exceeding $\tau = 6$. The number of false positives under various sampling periods ($1/f_s$, representing the sampling period) and the levels of pulsar faintness (represented by Γ) are shown in Tables A.7 and A.8 for SKA AA2 and MeerKAT telescope configurations, respectively.

According to the tables, brighter pulsars result in fewer false positives in non-pulsar snapshots. Compared to the results in Table 3, the numbers of false positives in non-pulsar snapshots are generally lower and therefore acceptable, even when the time difference between adjacent snapshots extends to 16 seconds.

4. Conclusions

In this article, we develop FITrig, a GPU-accelerated technique, for ULP pulsar detection and localisation. A ULP pulsar typically has a period longer than twice the sampling period of the telescope. Unlike traditional methods that perform a thorough search across entire images, FITrig offers advantages by increasing sensitivity to faint pulsars, reducing susceptibility to false positives from noise, and delivering substantial gains in computational efficiency through massive parallelisation, with the improvement being especially obvious for large images. In this article, the effectiveness of FITrig is validated on a real dataset to demonstrate its practicality and applicability, and FITrig is then tested on simulated datasets to showcase its capability over existing methods in more challenging cases.

The key novelties of FITrig include:

- **tLISI — a novel transient-oriented IQA index:** A statistically-grounded IQA metric specially designed to highlight pulsar-specific variations through probability theory, enabling more reliable transient detection.
- **Elimination of deconvolution:** FITrig completely removes the need for deconvolution in the detection process while maintaining accuracy, significantly reducing computational overhead.
- **Dual-domain detection framework:** Two branches of FITrig (image domain and image-frequency domain approaches) are developed, significantly improving the capability to detect faint ULP pulsars.

- **GPU-powered real-time processing:** FITrig is highly parallelised with the GPU implementation, enabling real-time processing and scalability for large datasets.

Looking ahead, future research could explore whether filling the sampling gaps between observational snapshots could further enhance detection sensitivity. Machine learning methods, such as image completion (Li et al., 2023), could be applied to complete the time-sequential pulsar images in the measurements. Furthermore, while this study focused on pulsar data (observed and simulated using radio telescope configurations), the application of FITrig may be expanded to a broader range of transients and wider observational bands in astronomy.

Acknowledgements

The authors acknowledge support received from the STFC Grant (ST / W001969 / 1). The authors also acknowledge the use of the University of Oxford Advanced Research Computing (ARC; Richards 2015) facility in carrying out this work. Special thanks are due to Vlad Stolyarov for helpful discussion about fast imaging. Gratitude is extended to Fred Dulwich and Ben Mort for valuable guidance on OSKAR. We wish to express our sincere thanks to Mike Giles for helpful knowledge and practice about CUDA programming learnt from Course on CUDA Programming on NVIDIA GPUs (<https://people.maths.ox.ac.uk/gilesm/cuda/index.html>). Sincere appreciation is extended to Jack White for his helpful discussion regarding time-domain transient detection.

References

- Adámek, K., Roy, J., Armour, W., 2022. A novel greedy approach to harmonic summing using GPUs. *Astronomy and Computing* 40, 100621.
- Andersen, B.C., Ransom, S.M., 2018. A Fourier domain ‘jerk’ search for binary pulsars. *The Astrophysical Journal Letters* 863, L13.
- Bezuidenhout, M.C., Bhat, N.D.R., Caleb, M., et al., 2025. Slow and steady: long-term evolution of the 76-s pulsar j0901-4046. *Monthly Notices of the Royal Astronomical Society* 540, 2131–2145. doi:10.1093/mnras/staf773.

- Bhardwaj, M., Gaensler, B.M., Kaspi, V.M., et al., 2021. A nearby repeating fast radio burst in the direction of M81. *ApJL* 910.
- Booth, R., Jonas, J., 2012. An overview of the MeerKAT project. *African Skies* 16, 101–104.
- Burke, B., Graham-Smith, F., Wilkinson, P., 2019. *An Introduction to Radio Astronomy*. 4 ed., Cambridge University Press, Cambridge.
- Caleb, M., Heywood, I., Rajwade, K., et al., 2022. Discovery of a radio-emitting neutron star with an ultra-long spin period of 76 s. *Nat Astron* 6, 828–836.
- Cornwell, T., Evans, K., 1985. A simple maximum entropy deconvolution algorithm. *A&A* 143, 77–83.
- Davidson, D., 2012. MeerKAT and SKA phase 1, in: *ISAPE 2012*, pp. 1279–1282.
- Desvignes, G., Kramer, M., Lee, K., et al., 2019. Radio emission from a pulsar’s magnetic pole revealed by general relativity. *Science* 365, 1013–1017.
- Dewdney, P., Turner, W., Millenaar, R., et al., 2013. *SKA1 System Baseline Design*. Technical Report SKA-TEL-SKO-DD-001. SKA Organisation. Macclesfield.
- Dewdney, P., Wagg, J., Braun, R., Turner, W., 2016. *SKA1-LOW Configuration - Constraints & Performance Analysis*. https://www.skao.int/sites/default/files/documents/d17-SKA-TEL-SKO-0000557_01_-DesignConstraints-1.pdf.
- Dulwich, F., Mort, B., Stolyarov, V., et al., 2022. *OxfordSKA/OSKAR*. doi:10.5281/zenodo.5722575.
- Hancock, P.J., Murphy, T., Gaensler, B.M., et al., 2012. Compact continuum source finding for next generation radio surveys. *Monthly Notices of the Royal Astronomical Society* 422, 1812–1824.
- Hoffmann, B., 1968. Pulsars and a possible new test of general relativity. *Nature* 218, 667–668.

- Hogbom, J., 1974. Aperture synthesis with a non-regular distribution of interferometer baselines. *Astron. Astrophys. Suppl.* 15, 417–426.
- Hopkins, A.M., Miller, C.J., Connolly, A.J., et al., 2002. A new source detection algorithm using the false-discovery rate. *AJ* 123, 1086–1094.
- Hou, X., Smith, D.A., Guillemot, L., et al., 2014. Six faint gamma-ray pulsars seen with the fermi large area telescope. *A&A* 570, 11.
- Jonas, J., 2009. MeerKAT — The South African array with composite dishes and wide-band single pixel feeds. *Proc. IEEE* 97, 1522–1530.
- Kirk, D., 2007. NVIDIA CUDA software and GPU parallel computing architecture, in: *Proceedings of the 6th International Symposium on Memory Management*, pp. 103–104.
- Law, C.J., Bower, G.C., Burke-Spolaor, S., et al., 2018. realfast: Real-time, Commensal Fast Transient Surveys with the Very Large Array. *ApJS* 236.
- Li, X., 2024a. Fast imaging trigger v1.0.0. doi:10.5281/zenodo.13901829.
- Li, X., 2024b. Image quality assessment-based fast imaging pipeline for transient detection with GPU acceleration. Ph.D. thesis. University of Oxford. doi:10.5287/ora-g7nnok9o9.
- Li, X., Adámek, K., Armour, W., 2023. GPU accelerated singular value thresholding. *SoftwareX* 23, 101500.
- Li, X., Adámek, K., Armour, W., 2024a. GPU accelerated image quality assessment-based software for transient detection, in: *Astronomical Data Analysis Software and Systems (ADASS) XXXIV*.
- Li, X., Adámek, K., Armour, W., 2024b. Intensity-sensitive quality assessment of extended sources in astronomical images. *ApJS* 274, 37.
- Li, X., Armour, W., 2022. Intensity-sensitive similarity indexes for image quality assessment, in: *2022 26th International Conference on Pattern Recognition (ICPR)*, pp. 1975–1981.
- Lorimer, D., Kramer, M., 2004. *Handbook of Pulsar Astronomy*. Cambridge Observing Handbooks for Research Astronomers, Cambridge University Press.

- Offringa, A.R., McKinley, B., Hurley-Walker, et al., 2014. WSClean: an implementation of a fast, generic wide-field imager for radio astronomy. *MNRAS* 444, 606–619.
- Okamoto, I., Sigalo, F.B., 2006. Pulsar Magnetohydrodynamic Winds. *Publications of the Astronomical Society of Japan* 58, 987–1013.
- Pelisoli, I., Marsh, T., Buckley, D., et al., 2023. A 5.3-min-period pulsing white dwarf in a binary detected from radio to X-rays. *Nat Astron* 7, 931–942.
- Ransom, S.M., Eikenberry, S.S., Middleditch, J., 2002. Fourier techniques for very long astrophysical time-series analysis. *The Astronomical Journal* 124, 1788.
- Richards, A., 2015. University of Oxford Advanced Research Computing. doi:10.5281/zenodo.22558.
- Serra, P., Oosterloo, T., Morganti, R., et al., 2012. The ATLAS3D project – XIII. Mass and morphology of H i in early-type galaxies as a function of environment. *Monthly Notices of the Royal Astronomical Society* 422, 1835–1862.
- Tchekhovskoy, A., Spitkovsky, A., Li, J.G., 2013. Time-dependent 3D magnetohydrodynamic pulsar magnetospheres: oblique rotators. *Monthly Notices of the Royal Astronomical Society: Letters* 435, L1–L5.
- Thompson, A., Moran, J., Swenson, G., 2017. *Interferometry and Synthesis in Radio Astronomy*. 3 ed., Wiley, Oxford.
- Thompson, A.R., Clark, B.G., Wade, C.M., Napier, P.J., 1980. The Very Large Array. *The Astrophysical Journal Supplement Series* 44, 151–167. doi:10.1086/190688.
- Trott, C.M., Wayth, R.B., Macquart, J.P.R., Tingay, S.J., 2011. Source detection in interferometric visibility data. I. Fundamental estimation limits. *The Astrophysical Journal* 731, 81.
- Urpin, V., 2011. Magnetohydrodynamic waves in the pulsar magnetosphere. *A&A* 535, L5.

- Wang, P., Li, D., Clark, C.J., et al., 2021. Fast discovery of an extremely radio-faint millisecond pulsar from the fermi-lat unassociated source 3fgl j0318.1+0252. *Science China Physics, Mechanics & Astronomy* 64.
- Westmeier, T., Kitaeff, S., Pallot, D., et al., 2021. sofia 2 - an automated, parallel H i source finding pipeline for the WALLABY survey. *Monthly Notices of the Royal Astronomical Society* 506, 3962–3976.

Appendix A. Appendix

- **Table A.5:** Abbreviations and definitions of technical terms used throughout the article.
- **Table A.6:** Sensitivity indicated by the z-score of the pulsar tile with the threshold of $\tau = 6$, when applying FITrig to time sequential snapshots.
- **Table A.7:** Number of false positives detected by FITrig in non-pulsar snapshots simulated using the SKA AA2 configuration.
- **Table A.8:** Number of false positives detected by FITrig in non-pulsar snapshots simulated using the MeerKAT configuration.

| Term | Abbreviation | Description |
|---|--------------|--|
| (Celestial) Transient | - | A radio source whose emission varies noticeably over time in observations. |
| Compute Unified Device Architecture | CUDA | A parallel computing platform and programming model developed by NVIDIA. |
| Fast Imaging | FI | Technique that reconstructs images from visibilities collected over short timescales and identifies transient candidates through statistical analysis of these images. |
| Fast Imaging Trigger | FITrig | An image-based transient detection technique demonstrating high performance. |
| Image Quality Assessment | IQA | Metric to evaluate the quality of images. |
| Dirty Beam | - | The point spread function (PSF) of the telescope array resulting from its incomplete <i>uv</i> -coverage. |
| Steady Features | - | The structures formed by steady sources together with the dirty beams convolved on them. |
| Steady Source | - | A radio source whose emission remains largely constant, showing no significant variability during observations. |
| Sky Brightness Distribution | SBD | Spatial distribution of brightness across the sky. |
| Transient-Oriented Low-Information Similarity Index | tLISI | Metric to quantify similarity in data with minimal information, focused on transient events. |
| Ultra-Long-Period Pulsars | ULP Pulsars | Pulsars with periods longer than twice the sampling period of the telescope. |

Table A.5: Abbreviations and definitions of technical terms used throughout the article.

| Z-score of pulsar tile | | | Γ | | | | | | | |
|------------------------|----------------|----------------|-------------|-------|-------|-------|-------|-------|------|---|
| $f_s/(2 \times f_p)$ | Telescope | L | 0.05 | 1 | 20 | 40 | 60 | 80 | 100 | |
| 0.0982 | MeerKAT | 258 | 63.72 | 63.08 | - | - | - | - | - | |
| | | 514 | 63.76 | 63.08 | - | - | - | - | - | |
| | | 1026 | 63.77 | 63.23 | - | - | - | - | - | |
| | | 2050 | 63.81 | 63.42 | - | - | - | - | - | |
| | SKA AA2 | 258 | 77.57 | 77.05 | - | - | - | - | - | |
| | | 514 | 77.53 | 77.26 | - | - | - | - | - | |
| | | 1026 | 77.49 | 77.04 | - | - | - | - | - | |
| | | 2050 | 77.47 | 76.76 | - | - | - | - | - | |
| | 0.9817 | MeerKAT | 258 | 63.52 | 63.68 | 7.79 | - | - | - | - |
| | | | 514 | 63.53 | 63.72 | 7.92 | - | - | - | - |
| | | | 1026 | 63.54 | 63.79 | 8.64 | - | - | - | - |
| | | | 2050 | 63.57 | 63.94 | 8.79 | - | - | - | - |
| SKA AA2 | | 258 | 76.49 | 76.48 | 13.07 | - | - | - | - | |
| | | 514 | 76.49 | 76.46 | 15.32 | - | - | - | - | |
| | | 1026 | 76.45 | 76.29 | 18.81 | - | - | - | - | |
| | | 2050 | 76.44 | 75.97 | 15.76 | - | - | - | - | |
| 9.8175 | | MeerKAT | 258 | 63.43 | 63.61 | 52.53 | 21.15 | 9.59 | - | - |
| | | | 514 | 63.43 | 63.60 | 52.54 | 21.10 | 9.50 | - | - |
| | | | 1026 | 63.44 | 63.62 | 53.15 | 21.79 | 9.84 | - | - |
| | | | 2050 | 63.46 | 63.65 | 51.95 | 20.13 | 8.97 | - | - |
| | SKA AA2 | 258 | 76.22 | 76.88 | 69.43 | 35.87 | 17.48 | 9.94 | 6.31 | |
| | | 514 | 76.21 | 76.85 | 71.00 | 41.00 | 20.73 | 11.89 | 7.59 | |
| | | 1026 | 76.20 | 76.80 | 71.29 | 43.62 | 22.70 | 13.17 | 8.46 | |
| | | 2050 | 76.21 | 76.72 | 70.46 | 42.94 | 22.33 | 12.98 | 8.35 | |

Table A.6: Sensitivity indicated by the z-score of the pulsar tile with the threshold of $\tau = 6$, when applying FITrig to time sequential snapshots.

| # False positives | $1/f_s$ (sec) | | | | | | | | | | | | | | | |
|-------------------|---------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Γ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 0.05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20 | 3 | 1 | 2 | 6 | 2 | 9 | 12 | 9 | 19 | 27 | 24 | 26 | 21 | 19 | 25 | 21 |
| 40 | 15 | 16 | 6 | 17 | 11 | 16 | 22 | 23 | 24 | 33 | 26 | 29 | 26 | 22 | 27 | 27 |
| 60 | 19 | 23 | 10 | 18 | 13 | 16 | 23 | 23 | 27 | 33 | 29 | 29 | 27 | 22 | 28 | 27 |
| 80 | 19 | 26 | 10 | 18 | 13 | 17 | 23 | 25 | 27 | 33 | 29 | 29 | 27 | 22 | 28 | 27 |
| 100 | 20 | 26 | 10 | 20 | 13 | 17 | 23 | 25 | 27 | 33 | 29 | 29 | 27 | 22 | 28 | 27 |

Table A.7: Number of false positives detected by FITrig in non-pulsar snapshots simulated using the SKA AA2 configuration.

42

| # False positives | $1/f_s$ (sec) | | | | | | | | | | | | | | | |
|-------------------|---------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Γ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 0.05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20 | 1 | 1 | 0 | 3 | 0 | 3 | 3 | 1 | 4 | 12 | 4 | 13 | 5 | 14 | 16 | 11 |
| 40 | 20 | 17 | 0 | 20 | 23 | 14 | 18 | 14 | 30 | 28 | 28 | 35 | 27 | 28 | 37 | 39 |
| 60 | 34 | 30 | 0 | 27 | 31 | 19 | 25 | 25 | 37 | 32 | 30 | 40 | 34 | 30 | 41 | 43 |
| 80 | 35 | 34 | 12 | 27 | 35 | 21 | 27 | 26 | 39 | 33 | 30 | 40 | 34 | 31 | 42 | 44 |
| 100 | 38 | 34 | 20 | 28 | 36 | 24 | 27 | 26 | 40 | 33 | 31 | 40 | 35 | 31 | 42 | 44 |

Table A.8: Number of false positives detected by FITrig in non-pulsar snapshots simulated using the MeerKAT configuration.