Supplementary Material for SIGNN – Star Identification using Graph Neural Networks

Floyd Hepburn-Dickins[1], Mark W. Jones[1], Mike Edwards[1], Jay Paul Morgan[1], Steve Bell[2]
[1]Swansea University, United Kingdom
[2]UK Hydrographic Office, Taunton, Somerset, United Kingdom
{880940,m.w.jones,Michael.Edwards,J.P.Morgan}@swansea.ac.uk, Steve.Bell@UKHO.gov.uk

1. Result Extraction

The comparison group consists of deep learning techniques of Spider-image CNN [3], RPNet [5] and classical technique grid algorithm [4]. Code implementations are unavailable for these techniques. Therefore, the results have been extracted from the figures reporting their results and conditions have been matched for testing. Specifically, the Spider-image CNN and RPNet results are from their respective papers [3,5]. The results used for the grid algorithm are those reported by RPNet [5] in their results and figures. The results were extracted from the result plots using the plot digitizer web tool [1] and the data used to plot new graphs to check they match the originals before use in this work.

2. Results Overview

The model is trained on data augmented with noise types of false star, dropped star, positional and magnitude. For initial submission, the model was trained to magnitude 5.0. In response to reviewers' comments to make the results more comparable, it was trained to magnitude 6.0. The magnitude 6.0 results for false star noise and dropped star noise appear in the main paper [2], and for the remaining noise types in this supplementary material as further explanation is required which exceeds the page limitation. This supplementary material also provides the magnitude 5.0 results and analysis.

The magnitude 5.0 results fulfill our goal of making a star identification model for a general cheaper camera which would have a wider field of view and be able to detect only the brighter stars (< 5 magnitude). The magnitude 6.0 model allow us to provide a fair comparison to current state-of-the-art models.

3. Converting Positional Noise to Angular Distance

The comparison group provides results on positional noise, where the noise is created by moving stars by a random number of pixels in a simulated image. Each sensor will have a different field of view, pixel density and lens distortion meaning that moving a star in units of a pixel will result in different movement in angular distances between sensors. We recommend that future work use angular distances to make comparisons easier to replicate.

In comparative works, results are reported as a standard deviation in pixel space, reflecting a training and testing process tied to the simulated sensor properties. That approach leads to a model that becomes specific to the simulated sensor, limiting its applicability to other configurations.

To compare position tolerance to the comparison group we converted each method's reported pixel noise results into a common angular distance frame. This is done by calculating the degrees-per-pixel of each method according to their simulated platforms by dividing their simulated sensor field of view (FOV) by it's resolution.

A comparison of sensor setups in the comparison group is summarized in Table 1.

Our proposed method of training and testing in the angular coordinate system avoids this issue, as angular distance tolerances can be easily applied across different sensors and other methods that also work in angular space.

4. Magnitude noise motivation

In an astronomical image, the magnitude M of a star is derived from its flux F using the following equation:

$$M = -2.5 \log_{10}(F) + C \tag{1}$$

where M is the magnitude, F is the flux, and C is a constant based on the reference flux. Whilst magnitude is a logarithmic measure of flux, many sources (dark current, background light pollution, read noise) of noise in an image act additively on the flux, rather than proportionally. This introduces a constant amount of uncertainty in the flux, which disproportionately affects fainter stars. These faint

Comparison of Simulation Platforms					
	FOV	**Pixel Size**	**Resolution**	**Max Magnitude**	**Approx degrees per Pixel**
RPNet	20°x 20°	0.012 mm	1024 x 1024 pixels	6.0 Mv	0.019531°
Spider-image CNN	20°x 20°	0.055 mm	2048 x 2048 pixels	6.0 Mv	0.009766°

Table 1. Comparison of simulation setups of RPNet [5], Spider-CNN [3].

stars, having lower initial flux values, are more susceptible to these disturbances and experience a greater relative change in observed brightness compared to brighter stars.

To model this behavior in SIGNN we apply magnitude noise as a percentage modifier to the original magnitude. Our approach effectively captures how noise tends to influence faint stars more significantly than bright stars, ensuring that the simulation better reflects real-world astrophotographic conditions. SIGNN will generally be more sensitive to magnitude noise, when compared to methods that use magnitude only as a filter for star visibility, due to its use of relative magnitude as a feature in the star pattern itself.

However, when considering real images, it is rare for nearby groupings of stars magnitude to fluctuate considerably in both degree and different directions. The use of relative magnitude ensures that, as long as the local brightness in a star-pattern is affected evenly then there will be negligible impact on SIGNN.

The performance of SIGNN towards magnitude noise is in table 2. In this testing, noise is created as random percentage modifier per node, and applied randomly as either a reduction or increase to it's magnitude. This can result in large shifts in the relative magnitudes between nearby stars. If the new magnitude is too faint for the simulated sensor to detect the node will be dropped. SIGNN accuracy drops to 76% under the highest level of this noise.

5. SIGNN 6.0 Results

The full results for SIGNN 6.0 are presented in Table 2. These include magnitude noise and dropped noise (percentage) that were not presented in the main paper. The celestial graph created for SIGNN 6.0 contains 4274 nodes and 20,310 edges. We test from noise levels of 5% for and at more granular levels of noise than the comparison group, and as such the expanded results are presented here. Training, including the parametric data-generation, took 16 hours on an Nvidia RTX 3090. Test images, simulated at 20x20°FOVs, take $\approx$ 5ms to predict all stars per image.

6. SIGNN 5.0 (Wide-FOV Model) Results

Table 3 shows SIGNN 5.0, a model trained towards use on wide-angle FOVs with a lower magnitude sensitivity. In marine navigation, long exposure times are impractical due to image degradation caused by the rocking motion and instability of the vessel. Additionally, cheaper, non-specialist sensors are more common. This requires a lower magnitude threshold (limiting the detection of fainter stars) and larger FOVs to capture sufficient information in each image. SIGNN 5.0 uses a connection radius of 10° and magnitude threshold of 5.0 to reflect this. The celestial graph created with these parameters contains 1645 nodes and 25,547 edges. The model architecture of SIGNN 5.0 is in figure 1.

References

[1] Plotdigitizer - web based online tool to extract data from plots, images and maps, 2024. Accessed: 2024-09-09. 1

[2] Floyd Hepburn-Dickins, Mark W. Jones, Mike Edwards, Jay Paul Morgan, and Steve Bell. SIGNN – Star identification using graph neural networks. In *accepted to Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, February 2025. 1

[3] Jie Jiang, Lei Liu, and Guangjun Zhang. Star identification based on spider-web image and hierarchical cnn. *IEEE Transactions on Aerospace and Electronic Systems*, 56(4):3055–3062, 2020. 1, 2

[4] C. Padgett and K. Kreutz-Delgado. A grid algorithm for autonomous star identification. *IEEE Transactions on Aerospace and Electronic Systems*, 33(1):202–213, 1997. 1

[5] Likai Xu, Jie Jiang, and Lei Liu. Rpnet: A representation learning-based star identification algorithm. *IEEE Access*, 7:92193–92202, 2019. 1, 2

Performance towards False Noise (% False stars added										
	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%
SIGNN	0.997	0.988	0.977	0.963	0.951	0.935	0.921	0.903	0.885	0.866
Performance towards Dropped Noise (%Stars dropped)										
	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%
SIGNN	0.995	0.987	0.976	0.962	0.944	0.923	0.890	0.854	0.819	0.762
Performance towards Position Noise (Std Dev in Radians)										
	0.0001	0.0002	0.0003	0.0004	0.0005	0.0006	0.0007	0.0008	0.0009	0.0010
SIGNN	0.998	0.998	0.998	0.997	0.998	0.997	0.996	0.996	0.995	0.994
Performance towards Magnitude Noise										
	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
SIGNN	0.999	0.997	0.994	0.992	0.987	0.979	0.959	0.916	0.852	0.760

Table 2. Full results for SIGNN trained to magnitude 6.0. Testing images simulated using a 10°radius. 800,000 (20,000 at each noise level) images were tested giving ≈ 28 million star predictions. Details on noise generation are available in the main paper.

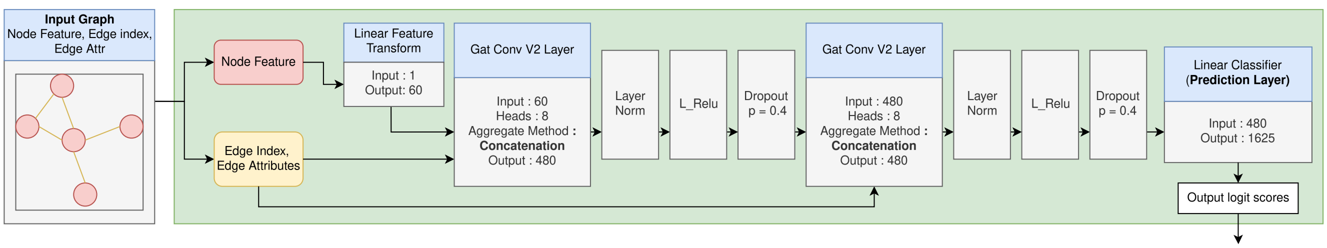

Figure 1. SIGNN 5.0 Model architecture for star identification using GAT layers.

Performance towards False Noise (% False stars added)										
	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%
SIGNN (5.0)	0.999	0.995	0.991	0.986	0.982	0.976	0.972	0.965	0.959	0.951
Performance towards Dropped Noise (% Stars dropped)										
	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%
SIGNN (5.0)	0.999	0.998	0.995	0.988	0.975	0.954	0.917	0.858	0.776	0.677
Performance towards Position Noise (Std Dev in Radians)										
	0.001	0.002	0.003	0.004	0.005	0.006	0.007	0.008	0.009	0.010
SIGNN (5.0)	0.999	0.999	0.999	0.998	0.998	0.997	0.996	0.993	0.990	0.987
Performance towards Magnitude Noise										
	1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
SIGNN (5.0)	0.999	0.997	0.994	0.992	0.987	0.979	0.959	0.916	0.852	0.760

Table 3. Full results for SIGNN 5.0. Testing images were simulated using a 40° radius. 400,000 (10,000 at each noise level) images were tested giving ≈ 70million star predictions. Details on noise generation are available in the main paper.