International Journal of Aviation Science and Engineering
e-ISSN: 2715-6958 p-ISSN: 2721-5342
AVIA

Original Article

A feature extraction method for star identification algorithm based on convolutional neural network

L Fathurrohim[1,*], B M Catraguna[1], and R E Poetro [1]

[1] Department of Aerospace Engineering, Institut Teknologi Bandung, Jalan Ganesha No. 10 Bandung 40132, Indonesia
* Correspondence: luqman.f@itb.ac.id

Received: 2 February 2024; Accepted: 30 April 2024; Published: 10 June 2024

The need to determine the orientation while in "Lost-In-Space (LIS)" is essential for spacecraft navigation. Star pattern recognition, also known as the star identification algorithm, plays a vital role for a spacecraft in LIS mode. Data-driven solutions for this type of problem are becoming more captivating due to their stochastic nature. This paper presents an efficient feature extraction method for the LIS star identification algorithm using a convolutional neural network. The net pattern and the multi-triangles feature extraction methods are implemented on the model. The proposed idea is tested on several simulated star images having a field of view of 25 by 16 degrees. The obtained results show an improvement in the successful identification rate of star image classes. Furthermore, the algorithm shows promising running time and requires less onboard memory since it eliminates storing a star catalogue for the matching process.

Keywords: orientation; spacecraft navigation; feature extraction; convolutional neural network; star catalog

1. Introduction

Star tracker is widely used in spacecraft to identify attitude determination by recognizing stars in the field of view (FOV). In Lost in Space (LIS), the star tracker works without any prior information of attitude is available. Methods for recognizing the detected stars are based on evaluating some relative positions between stars [1]. In this case, the star tracker must perform a full-sky star identification algorithm [2].

Star identification in LIS mode using a data-driven model has been widely used in recent years. Dos Santos et al. [3] used Convolutional Neural Network (CNN) with PointNet based architecture to identify the attitude of the star image. Rijlaarsdam et al. [4] utilized a deep Artificial Neural Network (ANN) for star identification algorithm with additional binary ranges feature extraction is implemented. Bobrovsky et al. [5] were successfully detected objects on star sky images by using CNN. Moreover, the work from [6] depicted an improvement of accuracy of CNN based model by enhancing an additional feature extraction.

The work aims to develop a feature extraction model that will increase the capabilities of CNN model to identify a guide star from a star image. The proposed model was trained with ten guide stars to compare the accuracy. The best performance of feature extraction model was also implemented to identify a star having a magnitude of brightness that is higher than 4.0 Mv.

2. Algorithm description

In this section, a description of the guide star identification is given. The proposed guide star identification algorithm consists of star image generation based on guide star position, feature extraction method, and CNN-based guide star identification.

2.1. Star Image Generation

The training dataset must be constructed to make any identification possible. The Smithsonian Astrophysical Observatory (SAO) star catalogue with the highest visual magnitude of 6.0 Mv is considered as the visible stars. Moreover, the highest visual magnitude of 4.0 Mv is chosen as the guide stars. The guide stars along with the visible stars around them must be determined to form classes of image. This is done by searching the stars catalogue with the right ascension (α) and declination (δ) that satisfy the following conditions

$$\alpha \in \left(\alpha_0 - \frac{R}{\cos\delta_0}, \alpha_0 + \frac{R}{\cos\delta_0}\right) \tag{1}$$

$$\delta \in (\delta_0 - R, \delta_0 + R) \tag{2}$$

$$R = \frac{1}{4}\left(FOV_x^2 + FOV_y^2\right)^{1/2} \tag{3}$$

Where α_0, δ_0, FOV_x, and FOV_y are the right ascension of the guide star, the declination of the guide star, horizontal field of view, and vertical field of view, respectively. After obtaining the list of stars that satisfy the conditions mentioned above, for each the stars' celestial coordinate system must be transformed to the image pixel coordinates using the following equations:

$$\begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} = M^T \begin{pmatrix} \cos\alpha_i \cos\delta_i \\ \sin\alpha_i \cos\delta_i \\ \cos\delta_i \end{pmatrix} \tag{3}$$

$$\begin{pmatrix} X_{pixel} \\ Y_{pixel} \end{pmatrix} = \frac{f}{z_i \mu} \begin{pmatrix} x_i/z_i \\ y_i/z_i \end{pmatrix} \tag{4}$$

Where M, f, and μ are a rotation matrix from the star sensor coordinate system to the celestial coordinate system, the focal length of the simulated camera, and pixel size of simulated camera respectively as explained in [7]. Random background noise is added to the simulated star images using the OpenCV's *addWeighted* module. The specifications of the simulated camera are shown in table X.

Table 1. Simulated camera specifications.

Specifications	Value
Pixel size	1.12 µm x 1.12 µm
Focal length	3.04 mm
Horizontal field of view	15.55 degree
Vertical field of view	12.2 degree

The star image simulator generates 480 classes of images based on the number of guide stars. To evaluate the performance of different feature extraction methods, we will use only 10 classes to reduce the computational resource needed for the training process.

2.2. Star identification Model

The star identification based on convolutional neural networks is used to evaluate the performance of feature extraction methods. The star identification algorithm uses to find the orientation of the guide star. In this work, we use a simple CNN that consist of several layers, as shown in figure 3.

e-ISSN: 2715-6958 p-ISSN: 2721-5342

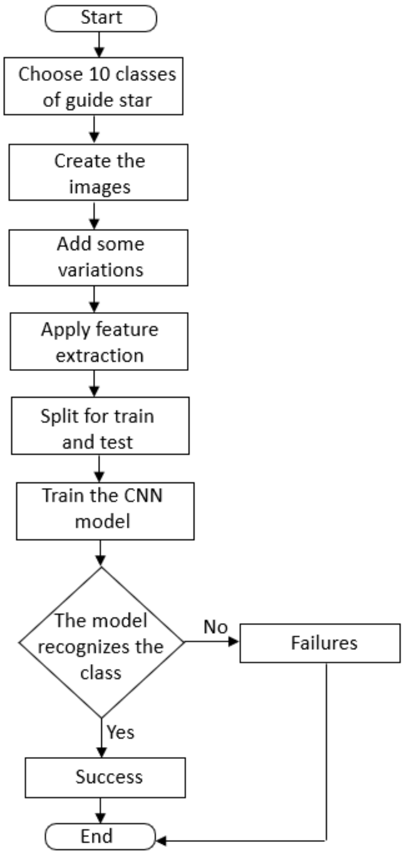

Figure 2. Flowchart of the evaluation of feature extraction method on the CNN model.

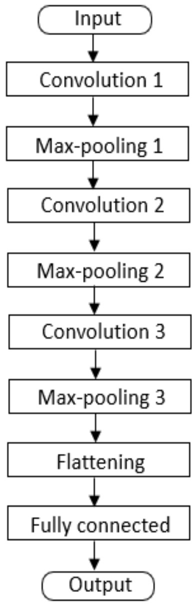

Figure 3. Flowchart of the evaluation of feature extraction method on the CNN model.

As shown in figure 3, the architecture of the network is described in detail in the following.

1. CONV layer. This layer receives an image as the input signal. There are 3 CONV layers with 3 x 3 convolutional kernels in the network. Each CONV layer uses a leaky rectified linear (ReLU) function as its activation function.
2. Pooling layer. Max-pooling is placed sequentially between each CONV layer. Max-pooling is performed with the maximum value over an input window by 2 x 2 and stride 2. After the last max-pooling layer, the output structures are flattened before coming to fully connected layers.
3. Fully Connected layer. After 3 CONV and pooling layers, three fully connected layers are used to classify the image. The fully connected layers consist of 256, 128, and 10 neurons. The first two neurons use ReLU as an activation function and SoftMax is used in the last layer.

The dataset was generated with the total number of images 145 in each class, where 20% was used as the testing dataset. The optimizer used was the rmsprop optimizer. The training process was done with 15 epochs.

2.3. Feature extraction method

The star image associated with a guide star has a unique distinction due to the nature of the star pattern. There are three different scenarios of evaluating star images for the CNN model. First, no feature extraction module is applied to the image. It will create a star pattern without any labelling. Second, we use a net-pattern label to the image. Third, a multi-triangles feature extraction module is applied to the image. A feature extraction module will enhance the input image such that it has an additional remark to be identified by the CNN model.

2.3.1. Net pattern feature extraction.

The net pattern feature extraction module generates a set of lines label between the guide star and n number of stars nearest the guide star.

Step 1: Select a guide star as the reference star S.
Step 2: Choose n other nearest star closest to the star S as adjacent stars A_i $(i = 1,2 \dots, n)$.
Step 3: Draw a line between star S to each adjacent star A_i.

2.3.2. Multi-triangles feature extraction.

The multi-triangles feature extraction module generates a set of triangles labels between the guide star and n number of stars nearest to the guide star.

Step 1: Select a guide star as the reference star S.
Step 2: Choose n other nearest star closest to the star S as adjacent stars A_i $(i = 1,2 \dots, n)$.
Step 3: Draw a line between all possible connection of star S and adjacent stars.

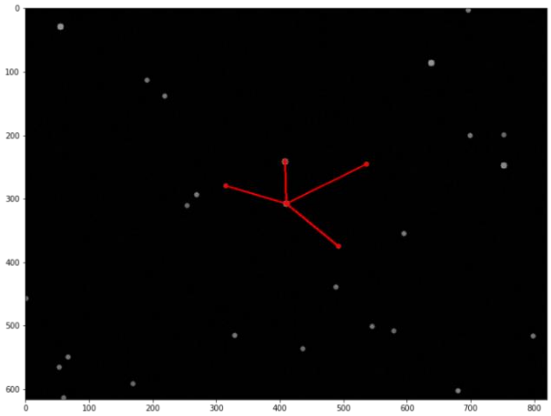

Figure 6. Net-pattern feature extraction method with $n = 4$.

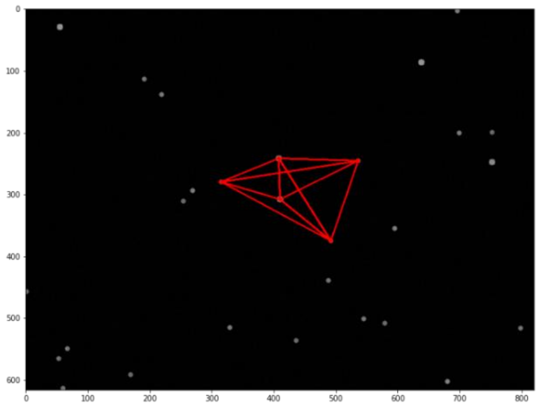

Figure 5. Multi-triangles feature extraction method with $n = 4$.

3. Simulation results and analysis

Evaluation of accuracy in the training process becomes the focus of this paper. The accuracy is analysed to understand the effectiveness of the feature extraction module to the CNN model. The accuracy of the guide star recognition is evaluated by following the standard procedures of a machine learning model where we split the dataset into training and testing data. With a ratio of 20% of the testing data, the trained CNN will be evaluated based on the testing data. If the CNN cannot predict the correct guide star based on the input image, the prediction is considered incorrect. Otherwise, it is correct. The accuracy can be retrieved by looking at the percentage of how much the CNN model can produce the right results.

First, the accuracy of the CNN model identifies the input image without any additional labelling is shown in figure 6. In this scenario, the accuracy of testing data in the CNN model is 0.75 at 15 epochs. The plot of loss vs. the number of iterations is shown in figure 7. Second, the accuracy of testing data in the CNN model increases to 0.88 at the number of iteration 15 after we apply net-pattern feature extraction as shown in figure 8. Third, the accuracy is slightly higher to 0.89 after 15 epochs after drawing a multi-triangles pattern on the input image as shown in figure 10. The value of loss vs. the number of epochs in applying net-patterns and multi-triangles feature extraction are shown in figure 9 and 11 respectively.

The multi-triangle feature extraction method, which has the highest accuracy in CNN model predicting 10 classes of star image, is applied to all guide stars having magnitude above 4.0 Mv. It can be seen from figure 11 that the model can achieve accuracy of 95%. Figure 12 shows the value of loss vs. the number of epochs of the model. The proposed model is shown to be flexible. The number of nodes in the layer can be adjusted to other number of output predictions and it is consistent with the work from [5].

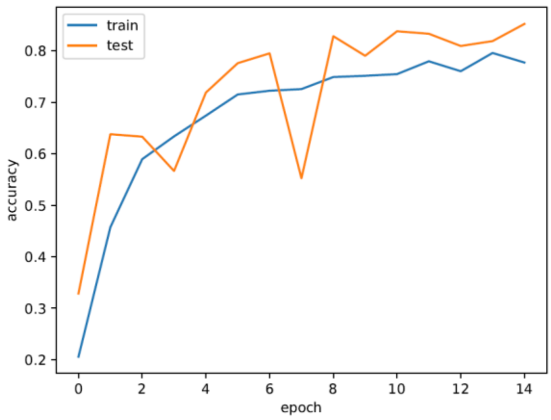

Figure 6. Accuracy vs. the number of epochs of CNN model predicted 10 classes without applying a feature extraction module.

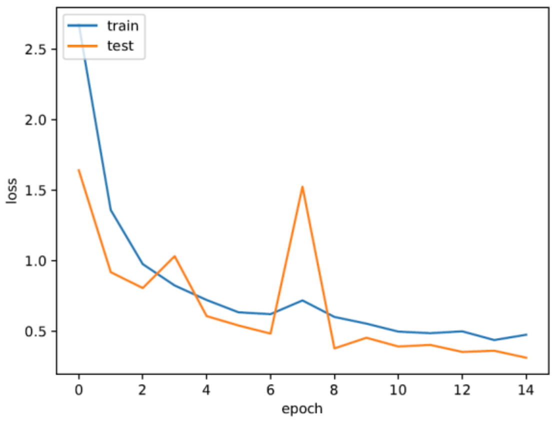

Figure 7. Loss vs. the number of epochs of CNN model predicted 10 classes without applying a feature extraction module.

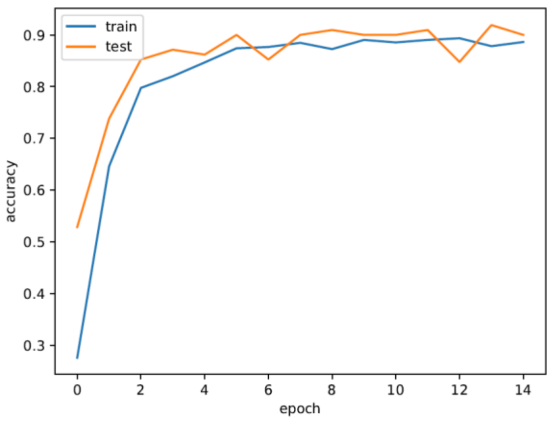

Figure 8. Accuracy vs. the number of epochs of CNN model predicted 10 classes with net-patterns feature extraction module.

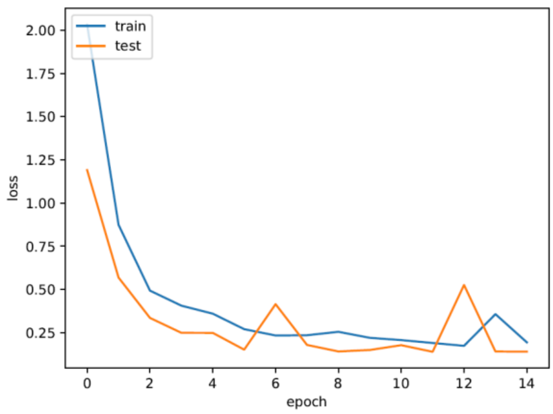

Figure 9. Loss vs. the number of epochs of CNN model predicted 10 classes with net-patterns feature extraction module.

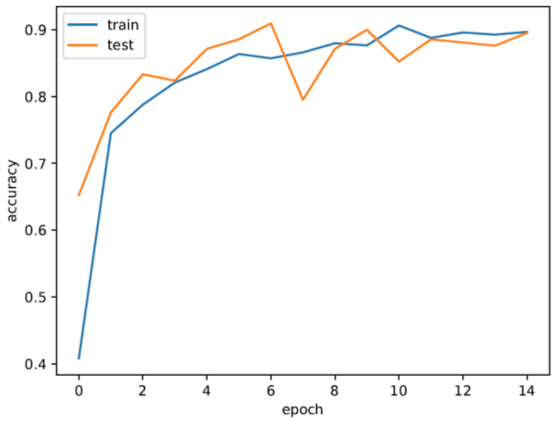

Figure 10. Accuracy vs. the number of epochs of CNN model predicted 10 classes with multi-triangles feature extraction module.

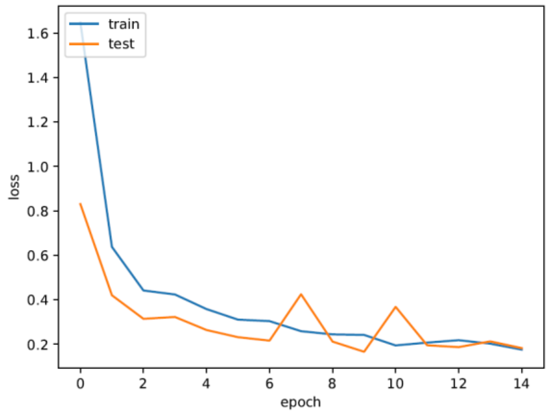

Figure 11. Loss vs. the number of epochs of CNN model predicted 10 classes with multi-triangles feature extraction module.

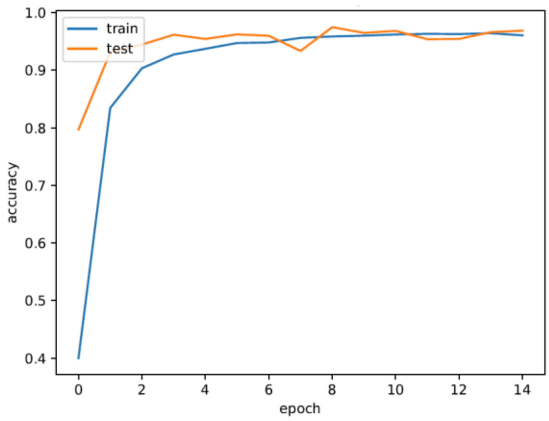

Figure 12. Accuracy vs. the number of epochs of CNN model predicted 480 classes with multi-triangles feature extraction module.

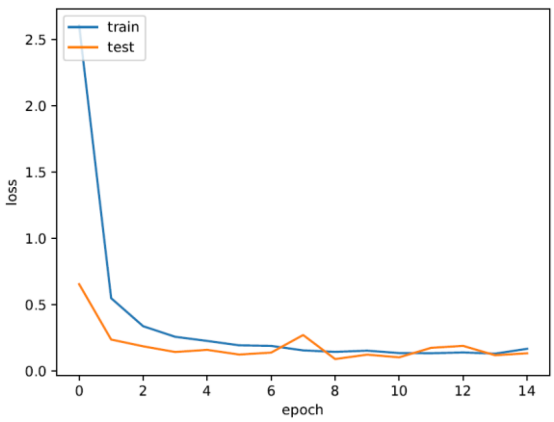

Figure 13. Loss vs. the number of epochs of CNN model predicted 480 classes with multi-triangles feature extraction module.

5. Conclusions

Applying feature extraction as a pre-processed image in star identification will increase the capability of a simple model of CNN to predict a star's position. Furthermore, net-pattern feature extraction and multi-triangles feature extraction will improve the accuracy of CNN by 13% and 14%, respectively, in the ten classes. However, the accuracy of CNN model with multi-triangles feature reached 95% in the training of all guide stars having a magnitude above 4.0 Mv. In the future, we will implement this feature extraction in the attitude determination algorithm in low-cost hardware for star sensors.

References

1. Mehta D S, Chen S, and Low K S 2018 A robust star identification algorithm with star shortlisting. Advances in Space Research, 61(10), pp.2647-2660.
2. Spratling B B and Mortari, D 2009 A survey on star identification algorithms Algorithms 2 (1), pp.93-107.
3. Dos Santos G H, Seman L O, Bezerra E A, Leithardt V R Q, Mendes A S, and Stefenon S F 2021 Static Attitude Determination Using Convolutional Neural Networks. *Sensors*, 21(19), p.6419.
4. Rijlaarsdam D, Yous H, Byrne J, Oddenino D, Furano G, and Moloney D 2020 Efficient Star Identification Using a Neural Network. *Sensors*, 20(13), p.3684.
5. Bobrovsky A I, Galeeva M A, Morozov A V, Pavlov V A, and Tsytsulin A K 2019 Automatic detection of objects on star sky images by using the convolutional neural network. *In Journal of Physics: Conf. Series* (Vol. 1236, No. 1, p. 012066). IOP Publishing.
6. Lee S J, Chen T, Yu L, and Lai C H 2018 Image classification based on the boost convolutional neural network. *IEEE Access*, 6, pp.12755-12768.
7. Ardi, N S, Saifudin M A, Poetro R E and Fathurrohim L 2018 Development of star image simulator for star sensor algorithm validation. *In Journal of Physics: Conf. Series* (Vol. 1130, No. 1, p. 012020). IOP Publishing.