PROCEEDINGS OF SPIE

SPIEDigitalLibrary.org/conference-proceedings-of-spie

Exoplanet detection from starshade images using convolutional neural networks

Zahra Ahmed, Simone D'Amico, Renyu Hu, Mario Damiano

Zahra Ahmed, Simone D'Amico, Renyu Hu, Mario Damiano, "Exoplanet detection from starshade images using convolutional neural networks," Proc. SPIE 12680, Techniques and Instrumentation for Detection of Exoplanets XI, 1268028 (5 October 2023); doi: 10.1117/12.2676600

SPIE.

Event: SPIE Optical Engineering + Applications, 2023, San Diego, California, United States

Exoplanet Detection from Starshade Images using Convolutional Neural Networks

Zahra Ahmed[a], Simone D'Amico[a], Renyu Hu[b], and Mario Damiano[b]

[a]Department of Aeronautics and Astronautics, Stanford University, Stanford, CA, 94305, USA
[b]Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109, USA

ABSTRACT

High-contrast, direct imaging of exoplanets is a key objective for the future of astronomy. Underpinning both the development and successful deployment of future direct imaging missions is the ability to analyze the resulting data. Therefore, in order to inform noise budget requirements and fully leverage high-resolution direct images, it is critical to develop post-processing algorithms that are versatile, efficient, and reliable. This work leverages recent advancements in starshade simulation technology and presents a Convolutional Neural Network (CNN) for detecting exoplanet signals from direct images. The proposed architecture modifies U-Net, a semantic segmentation architecture, to directly regress the coordinate location of exoplanet signals from a synthetic starshade imaging dataset. Results from a limited test set of 144 images indicate that the CNN achieves an average precision and recall of 0.925 ± 0.012 and 0.923 ± 0.018, respectively. Additionally, the CNN does not require processing the data through conventional background estimation algorithms prior to detection and performs inference on single images, showcasing the potential of supervised machine learning as a versatile approach for exoplanet detection.

Keywords: Exoplanet Detection, Direct Imaging, Starshades, Image Processing, Machine Learning, Convolutional Neural Networks

1. INTRODUCTION

Direct imaging can identify exoplanets and characterize their atmospheres through the acquisition of their spectra, thus providing key information about whether they are capable of supporting life. As such, the coming decades will see an increased focus on direct imaging missions to detect and characterize habitable worlds, including the Habitable Worlds Observatory (HWO) that has been identified as a result of recommendations from the Astro2020 Decadal Survey.[1] However, directly imaging an exoplanet is challenging due to light from the host star, which is orders of magnitude brighter than the planet and therefore washes out its signal. Even with advances in adaptive optics and starlight suppression technology, direct images have low signal-to-noise ratios (SNR) and require significant post-processing in order to be interpreted. Therefore, image processing algorithms are critical to the entire life cycle of future direct imaging missions because they are necessary both to validate the performance of novel mission concepts and to ensure that the data captured from these missions is processed accurately and efficiently.

The current state-of-the-art in exoplanet image processing, while accurate, has been primarily focused on calibration techniques designed for coronagraph images. The next generation of space-based, direct imaging missions will require exploring novel mission architectures where these existing techniques are less effective. Additionally, current image processing algorithms often rely on visual inspection as a final step in the detection pipeline. As the backlog of available direct imaging data continues to grow into the order of petabytes,[1] the human-in-the-loop dependence of existing algorithms will prevent them from efficiently processing these large datasets and thus possibly delay important exoplanet discoveries. Therefore, in order to fully harness the potential of direct imaging, new exoplanet detection algorithms must address the following three challenges: 1) accurately detect exoplanet signals in low-SNR direct images, 2) robustly evaluate the performance of diverse mission concepts, and 3) efficiently process large amounts of data.

Further author information: (Send correspondence to Z.A.)
Z.A. Email: zjahmed@stanford.edu

Techniques and Instrumentation for Detection of Exoplanets XI, edited by
Garreth J. Ruane, Proc. of SPIE Vol. 12680, 1268028 · © 2023 SPIE
0277-786X · doi: 10.1117/12.2676600

To address these challenges, this work approaches exoplanet image processing using supervised machine learning, specifically Convolutional Neural Networks (CNNs). CNNs represent end-to-end automated image processing algorithms that learn the features of the data during training. This allows them to efficiently evaluate large datasets while reducing prior assumptions once the model is trained. CNNs have been successfully applied to related detection problems in fields such as radio astronomy, spacecraft pose estimation, and biomedical image processing.[2–4] The diversity of these applications indicates that CNNs are a versatile approach that can be extended to exoplanet detection in starshade images.

To date, CNNs have had limited applications to exoplanet direct imaging due to a lack of training data. However, recent advances in synthetic starshade images through the Starshade Imaging Simulation Toolkit for Exoplanet Reconnaissance (SISTER)[5*] now enable the creation of labeled datasets that can be used to train CNNs and evaluate their accuracy on low SNR direct images. Starshades are a salient use case for this work because they represent a novel starlight suppression technology with the potential to provide higher image contrast than coronagraphs, with studies indicating that they can suppress starlight by a factor of 10^{10} at the inner working angle (IWA).[6,7] This performance could enable starshade missions to image previously inaccessible exoplanet candidates, including potential Earth-analogs, as well as to detect multiple planets in the same system. Several starshade missions have been proposed in recent years,[8–11] and the recent Starshade Exoplanet Data Challenge[12] highlights the necessity of novel image processing algorithms to inform noise budget requirements for future starshade missions.

To this end, this work leverages advancements in supervised machine learning as well as SISTER to develop CNNs for exoplanet detection. Specifically, the main contribution of this work is a novel CNN approach that modifies U-Net, a semantic segmentation architecture, to regress the location of exoplanet signals from a SISTER dataset. In this study, the exoplanet signals are treated as point sources and minimal expert knowledge of the data is integrated into the CNN training pipeline to establish a baseline that can be used as a comparison for future studies. Although starshades are the use case for this work, the presented methodology can be tailored to different direct imaging payloads, making it suitable for evaluating and comparing various mission concepts. Therefore, the impact of this work is two-fold. First, it helps enable novel mission concepts for direct imaging by investigating the next generation of versatile and efficient image processing algorithms. Second, it expands the application of machine learning in astronomy and astrophysics, which has been underexplored despite the increasing need to reduce post-processing bottlenecks as the amount of available data increases.

The remainder of this paper proceeds as follows. Section 2 provides an overview of the state-of-the-art in exoplanet direct imaging. Section 3 provides a brief background on CNNs before sections 4 and 5 present the dataset and CNN used in this work. Section 6 details the results from training the CNN and compares it to conventional approaches before Section 7 summarizes and provides the future direction of this work.

2. RELATED WORK

This section reviews the state-of-the-art in exoplanet image processing algorithms by categorizing them into two main areas: background subtraction pipelines and supervised machine learning approaches. Background subtraction pipelines are composed of two steps, background estimation and detection, each with distinct algorithms.

2.1 Background Estimation Algorithms

The majority of exoplanet image processing techniques have been focused on coronagraph images, where the primary objective is to calibrate and remove quasi-static speckle patterns that are caused by imperfections in the instrument. Speckle noise is correlated over time and wavelength, allowing it to be approximated by differential imaging techniques. Differential imaging is a high-contrast imaging technique that uses series of images captured at various roll angles in the case of Angular Differential Imaging (ADI),[13] or at various wavelengths as with Spectral Differential Imaging (SDI).[14] These ADI or SDI sequences can then be used to construct a reference point spread function (PSF) using algorithms such as LOCI,[15] KLIP,[16] or LLSG.[17] Following PSF subtraction, the images can be passed through separate detection algorithms. Thus, though these algorithms have been widely

*http://sister.caltech.edu

used for successfully confirming exoplanet candidates from coronagraph images, they are one part of a multi-step pipeline that is often not scalable to large datasets due to the reliance on visual inspection and statistical analyses at each step to refine planet predictions. Additionally, their dependence on differential imaging for optimal results makes them less suitable for space-based missions and starshades, where obtaining sequences of images is less feasible due to scheduling and fuel constraints.

2.2 Detection Algorithms

Similar to background subtraction algorithms, many detection algorithms have been focused primarily on coronagraph images and therefore are similarly reliant on ADI or SDI sequences. Cantalloube et al. developed ANDROMEDA, which uses a maximum likelihood approach coupled with ADI to automatically detect the position and flux of exoplanet point sources.[18] Ruffio et al. instead used a matched filter and forward modeling approach to detect exoplanet signals from background-subtracted images that are first passed through the KLIP algorithm.[19] This approach was incorporated into an open-source Python library known as pyKLIP,[20] which allows the user to implement the full pipeline from raw images to exoplanet detections.

More recently, Hu et al. developed a generalized likelihood ratio test (GLRT) approach for signal detection from starshade images.[12] This approach is the first dedicated algorithm tailored to starshade image processing and does not require any background estimation algorithms prior to detection or rely on differential imaging datasets. The GLRT method represents an automated starshade image processing algorithm that does not employ supervised machine learning. However, its computational complexity compared to supervised machine learning approaches has not yet been studied, particularly for its iterative application for images containing exozodiacal dust.

2.3 Supervised Machine Learning Approaches

In recent years, there have been a few supervised machine learning approaches to exoplanet image processing, all confined to coronagraph images. Gomez Gonzalez et al. employed both random forests and neural network techniques to identify exoplanets from ADI sequences of data.[21] Both algorithms were shown to outperform principal component analysis (PCA) based algorithms in true positive detection rates and additionally represent end-to-end algorithms. These algorithms highlight the promising nature of supervised machine learning over classical statistical techniques. However, similar to LOCI, KLIP, and ADROMEDA, the CNN developed by Gomez Gonzalez et al. relies on datasets collected using ADI and therefore cannot be applied to single images. Yip et al. identified the lack of supervised machine learning approaches for single images and developed a Generative Adversarial Network (GAN) to produce synthetic coronagraph images for training a CNN that can perform binary classification.[22] The binary classifier designates images as positive or negative but does not directly output the location of the signals in the image. Thus, there is still a gap in developing CNN architectures for locating exoplanet signals from single images that is addressed in this work.

3. CNN BACKGROUND

CNNs are a form of supervised learning, meaning that the true value of the information being regressed, known as the ground truth, is used to update the model during training. Implementing a CNN requires a labeled dataset, the CNN architecture, and a training pipeline.

A CNN's architecture consists of two main components: a feature extractor and a detection head. The feature extractor learns key patterns in a dataset of images through the use of sequential filters composed of learnable parameters. These filters apply convolution operations to the data and are combined with other types of layers such as pooling and non-linear activation functions to successively alter the resolution of the output. The output of the feature extractor is then fed into the detection head, which can be used to predict information about objects in the images, such as their location or class.

The primary objective of the training pipeline is to minimize a loss function by optimizing the parameters of the CNN. Therefore, loss functions are carefully selected to quantify the difference between the network output and ground truth. Thus, the training process of a CNN first involves passing a subset of data, called a batch, through the CNN and computing the loss. Then, back-propagation is used to compute the gradient of the loss

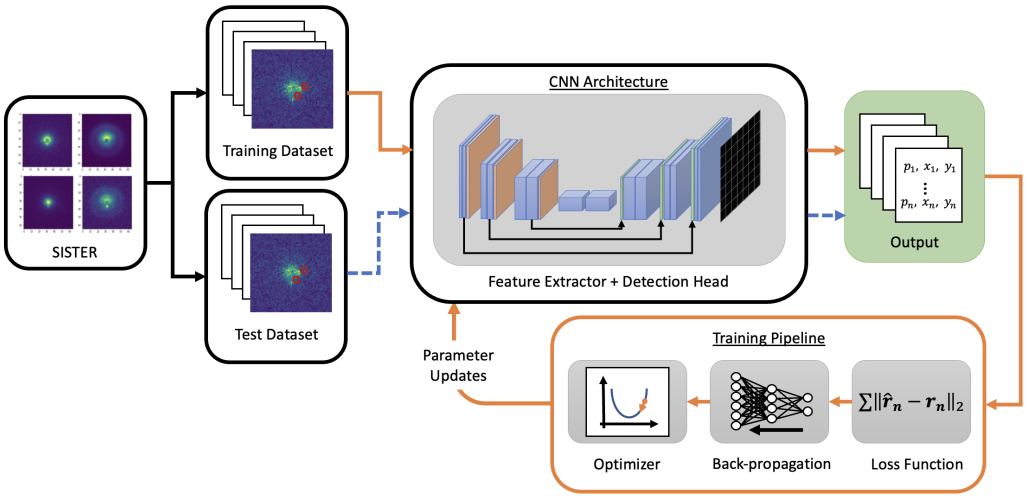

Figure 1: High-level overview of the CNN training and inference process using a SISTER dataset. The solid orange line indicates the training loop while the dashed blue line designates the forward pass of the model at inference, after training is complete. The details of the architecture and training process used in this work are discussed in Section 5.

function with respect to the network parameters. Finally, this gradient is fed into an optimization algorithm that determines how to update the parameters of the network. The training process is complete once the designated number iterations through the dataset, or epochs, have been completed and the loss has converged. After training, the CNN performs inference on a test dataset via a forward pass through the model, with the assumption that the test dataset is separate from the training dataset but generated from the same satistical distribution. The problem of making a CNN robust to test data that is out-of-distribution is an active field of research[23–26] and will be expanded upon in Section 6.6. An overview of the training and inference process for the CNN used in this work is shown in Figure 1.

4. SYNTHETIC STARSHADE DATASET

The dataset used in this work was generated using SISTER for the Starshade Exoplanet Data Challenge (SEDC).[27] It consists of 1,440 65×65 single-band images with an observatory model of the Starshade Rendezvous Probe that was proposed for the Nancy Grace Roman Space Telescope.

The high level details of parameters varied in the dataset are summarized in Table 1, however additional details on SISTER and how the dataset was generated can be found in Refs. 5, 27 and 28. The dataset models the geometric IWA of the Starshade Rendezvous Probe as 72 milli-arcseconds (mas) for the 425-552 nm passband and 104.3 mas for the 615-850 nm passband, consistent with the most recent design.[9] The starshade contrast is defined as the ratio of starlight at the IWA to starlight without the starshade, and is modeled at a nominal (10^{-10}) and degraded (10^{-9}) performance.[28] The dataset includes ten hypothetical exoplanetary systems that are associated with four stars, and each scenario is modeled for two epochs, or visits. Two to three planets within 2 Earth radius are assumed to exist in or near the habitable zones of the considered stars. The input parameters for each visit are then varied for every permutation of two imaging passbands, two models of exozodiacal dust at three density levels, and three reference planet SNRs. In total, these permuted simulations comprise the 1,440 images in the dataset and include several sources of astrophysical, detector, and instrument noise that are unique to starshade images. Several of these sources of noise, including dark current noise and shot noise are modeled assuming Poisson distributions, thereby adding additional variability to the dataset. Figure 2 shows how a single scenario varies with degraded starshade contrast and a resonant exozodiacal disk model.

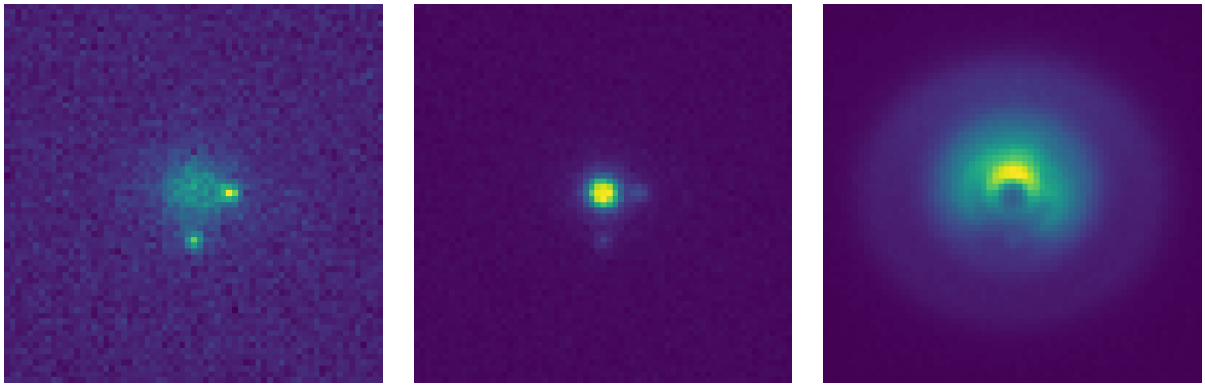

Figure 2: Three sample images from the SEDC dataset. All three images model the same artificial exoplanetary system associated with the star β Canum Venaticorum and are simulated in the blue (425 - 552 nm) passband. (*Left*) The scene with a smooth exozodi model with low density, SNR 20, and nominal starshade contrast of 10^{-10}. (*Middle*) The same scene with degraded starshade contrast, 10^{-9}. (*Right*) The same scene as on the left with a resonant exozodi model with high density. All three images are shown with a viridis colormap applied with varied color scales to highlight the features of each image.

The labels for the dataset were generated using the inputs of the simulations, which include the injected planets' location with respect to the star. These locations can be converted to camera pixels using the pixel pitch of the camera, which is defined as 21.85 mas per pixel. In this work, the exoplanet signals are treated as point sources; their spatial extent and PSF in the images is not considered.

CNNs require extensive datasets in order to accurately generalize to data that is not seen during training, typically on the order of thousands to tens of thousands of images. Therefore, the SEDC dataset can be considered a limited dataset in terms of quantity of images compared to what would generally be used to train a CNN. Generating additional realistic and meaningful images to supplement those in the SEDC dataset will require a concentrated effort that was not included in the scope of this feasibility study but will be addressed in future work. Instead, the effects of using limited training data were mitigated using common training techniques that are discussed in Section 5.3.

5. CNN ARCHITECTURE AND TRAINING

This section presents the CNN architecture and training pipeline used in this work. The approach builds off of U-Net, a semantic segmentation architecture that was originally developed for biomedical image processing.[4] Semantic segmentation CNNs assign each pixel in an image to a class with an associated confidence. This

Table 1: Parameters varied in the SEDC dataset, synthesized from Refs. 27 and 28.

Parameter	Value(s)
Starshade Contrast at IWA	10^{-10}, 10^{-9}
Imaging Passbands	425-552 nm, 615-850 nm
Exoplanetary Systems	10 systems
Stars	τ Ceti, ϵ Indi A, σ Dragonis, β CVn
Exozodi Models	resonant, smooth
Exozodi Density	1, 3, 10 zodis (τ Ceti: 3, 10, 30 zodis)
Idealized SNR	5, 10, 20
Visits	2

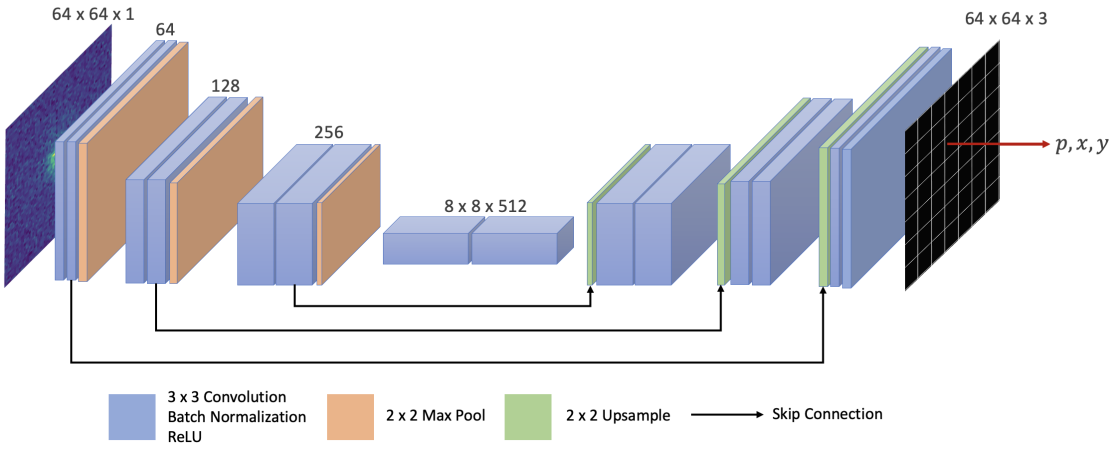

Figure 3: The modified U-Net architecture. The shape of the encoding pathway is mirrored by the decoding pathway, creating a "U-shaped" architecture. The final layer is modified to produce a three-channel output for each pixel.

pixel-wise information can be used to create a map of the input image that is segmented into regions based on the predicted classes. In its original implementation, U-Net was shown to provide accurate segmentation with relatively few training images, making it a suitable choice for this application.

Here, the original U-Net was modified in three main ways. First, the depth of the network was reduced to accommodate the small size of the input images. Second, the detection head was modified to directly regress the location of exoplanet signals in the image. Finally, the training loss was updated to account for both the location and confidence of the exoplanet predictions.

5.1 Architecture

The network takes in a 64×64 input that is first passed through a double convolution block, which consists of two 3×3 convolutions, each followed by batch normalization and a Rectified Linear Unit (ReLU) activation. It is then passed through an encoding pathway where it is downsampled three times using a 2×2 max pool operation, as opposed to the original U-Net which performs four downsampling operations. This modification follows the implementation of U-Net used by Ribera et al.[29] and is necessary due the small size of the input images. After each of the three downsampling operations, the input is again passed through a double convolution block with two 3×3 convolutions, batch normalization and a ReLU layer. Similar to the original U-Net implementation, the number of feature channels is doubled for each pair of convolutions, beginning with 64 channels and ending with 512 channels. The output of the encoding pathway is a $8 \times 8 \times 512$ feature map as shown in Figure 3.

The encoded feature map is then passed through a decoding pathway that successively increases the resolution of the output. At each upsampling operation, the input is upsampled using bilinear interpolation and then concatenated with the corresponding encoding feature map in what is referred to as a skip or shortcut connection. These skip connections allow information from earlier layers to be recovered in later layers, and have been shown to increase the stability of the training process.[30] After the skip connection, the input is passed through the same double convolution block as the encoding pathway, except with the feature channels decreasing back to 64.

The output of the expanding pathway is passed through the detection head, a 1×1 convolution layer which has been modified to have three channels instead of one. The three channels produce three logits, or real numbers between negative and positive infinity. The logits are then passed through a final nonlinear activation, in this case a sigmoid function which bounds the three values to floating point numbers between 0 and 1. The first value is interpreted as a confidence, p, for each prediction, representing a probability between 0 and 1. The last

two values are interpreted as (x, y) locations for each pixel, representing the fractional pixel value. Therefore, the final output of the CNN is pixel-wise predictions of p, x, y corresponding to proposed coordinate locations of exoplanet signal in fractional pixels with an associated confidence between 0 and 1. In total, this architecture has 6.02 million trainable parameters.

5.2 Training Loss

The loss function of a CNN must be differentiable with respect to the network output in order to compute the gradient via back-propagation. By modifying the output convolution layer of U-Net to explicitly predict a location and associated confidence for each pixel, the loss function can encode this location information and therefore more directly optimize the network for the intended task of exoplanet detection. This is in contrast to typical binary classification semantic segmentation approaches that exclusively output the probability of an object in that pixel, and then require clustering algorithms applied after inference to localize the center of the object from regions of high probability.

Here, the loss function was modified from the original U-Net implementation and is composed of two main parts. First, the mean squared error (MSE) is calculated between the location predictions ($\vec{r_i}$) and ground truth exoplanet signal locations ($\hat{\vec{r_i}}$), only for predictions that correspond to the same pixel as a ground truth signal. The predicted locations are the pixel-wise float outputs, (x, y), of the CNN that are then un-normalized by the location of the pixel in the image to represent the absolute location of the prediction in pixels, same as the ground truth locations. The MSE value is normalized by the integer number of ground truth signals for that batch (n) and is defined as,

$$L_{MSE} = \frac{1}{n} \sum_{i=1}^{n} \|\vec{r_i} - \hat{\vec{r_i}}\|_2^2 \tag{1}$$

Second, the focal loss[31] is used to fine-tune the confidence, p_i, of the prediction made for each pixel, which represents a probability between 0 and 1. The focal loss adds a modulating term, controlled by the parameter γ, to the binary cross entropy (BCE) loss. The BCE loss is defined as

$$L_{BCE} = -\frac{1}{n} \sum \hat{p_i} log(p_i) + (1 - \hat{p_i}) log(1 - p_i) \tag{2}$$

where $\hat{p_i}$ is the ground truth class. If a pixel contains an exoplanet signal, $\hat{p_i} = 1$; if there is no exoplanet signal $\hat{p_i} = 0$. In other words, the first term in the BCE loss corresponds to "positive" samples ($\hat{p_i} = 1$) and the second term corresponds to "negative" samples ($\hat{p_i} = 0$). The BCE loss is therefore minimized when predictions have a high confidence in pixels where there truly is an exoplanet signal, and low confidence for pixels where there is not. The focal loss is then defined as

$$L_{focal} = -\frac{1}{n} \sum_{i=1}^{BWH} \alpha (1 - p_i)^{\gamma} \hat{p_i} log(p_i) + (p_i)^{\gamma} (1 - \hat{p_i}) log(1 - p_i), \tag{3}$$

where γ represents the focal parameter, α is the positive sample weight, and n is the total number of ground truth samples. The focal loss is summed over all pixels in the batch, where a batch consists of a certain number of images (B), and each image contains a number of pixels defined by its width (W) and height (H).

The modulating term, $(1 - p_i)^{\gamma}$ or $(p_i)^{\gamma}$ in Eq. 3, decreases the contribution of predictions that are "well-classified," meaning their values are close to the ground truth class. Consequently, it amplifies the relative importance of predictions that are "misclassified," or further away from the ground truth class. For example, if a predication for a pixel where there is a ground truth signal ($\hat{p_i} = 1$) has a confidence close to 0, it contributes more to the focal loss than a prediction for the same pixel with a confidence near 1. The degree of this effect can be controlled by increasing γ.

Additionally, the positive sample weight, α, is used to address the class imbalance in the dataset. This is important because most pixels do not have an exoplanet signal. Thus, without α, the focal loss would largely

reflect only "negative" samples. To ensure the model properly learns to identify "positive" samples, α is used to increase the significance of these predictions in the loss computation. Both γ and α were manually tuned and set to final values of $\gamma = 2.0$ and $\alpha = 25$.

The MSE and focal loss are summed to compute the total loss used for backpropogation, such that

$$L_{total} = L_{MSE} + L_{focal}. \tag{4}$$

5.3 Training and Implementation Details

Before training, the images from the starshade dataset were cropped from 65×65 to 64×64 to fit the input size required by the CNN architecture. Additionally, 10% of the starshade dataset was randomly partitioned and set aside for testing. The remaining 90% was used for training and validation using k-fold cross validation, a resampling technique that is commonly used for limited datasets. In k-fold cross validation, the training data is randomly divided into k partitions. Then, k models are trained, each with a different partition held out for validation. In this work, $k = 8$ to ensure that each split had the same number of images. To further mitigate the effects of a limited dataset, the data was augmented during training using random vertical and horizontal flips and random rotations in increments of 90°. These geometric transformations help make the CNN more robust to the location of the signals in the image. The CNN was trained using the Adam optimizer[32] with a learning rate of 10^{-4}. The batch size was set to 16 images and each of the 8 models was trained for 200 epochs. The CNN and training pipeline were implemented in PyTorch v1.10.2 and trained using an NVIDIA GeForce RTX 2080 Ti 12GB GPU.

6. RESULTS

This section first details the inference pipeline that was used to make predictions on unseen data after the CNN was trained as well as describes the evaluation metrics used in this work. It then describes the results of the CNN obtained from k-fold cross validation and visualizes the learned features of the network. The results of the CNN are then compared to two conventional approaches for exoplanet detection. Finally, this section discusses the current limitations of this work in regards to the CNN's ability to generalize to out-of-distribution data.

6.1 Inference Pipeline

Once the CNN has been trained, the inference pipeline first involves a forward pass through the network, which produces a $64 \times 64 \times 3$ output map that is passed through a sigmoid function and provides a p, x, y, prediction for each pixel. The confidence value, p, is then used to threshold the predictions. For this work, the confidence threshold, τ, was set to $\tau = 0.5$. The hyperparameters for the training and inference pipelines are summarized in Table 2.

Table 2: Training and inference hyperparameters.

Parameter	Description	Value
γ	Focal loss parameter	2.0
α	Positive sample weight	25
τ	Confidence threshold	0.5

6.2 Evaluation Metrics

Three metrics were used to evaluate the performance of the CNN: precision, recall and mean L_2 error. The precision and recall were computed as

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

$$Recall = \frac{TP}{TP + FN}\,, \tag{6}$$

where TP refers to true positive detections, FP refers to false positive detections and FN refers to false negative detections. Precision specifies how dependable the predictions from the CNN are, while recall indicates how completely the CNN identifies all true signals. The definition of a true positive was established by the full-width at half maximum (FWHM) resolution of the Roman starshade, which is 65 mas at 750 nm.[33] This corresponds to a FWHM of approximately 37 mas at 425 nm, or 1.7 camera pixels, and 53 mas at 615 nm, or 2.4 camera pixels. Therefore, a true positive was conservatively defined as any detection ($\vec{r}_{TP}$) that was within two pixels of a ground truth signal ($\vec{r}_{GT}$), i.e.,

$$TP = \|\vec{r}_{TP} - \vec{r}_{GT}\|_2 \leq 2\ px\,. \tag{7}$$

The mean L_2 error was calculated for the true positive detections as

$$Mean\ L_2 = \frac{1}{n_{TP}} \sum_{i=1}^{n_{TP}} \|\vec{r}_{TP} - \vec{r}_{GT}\|_2\,, \tag{8}$$

where n_{TP} is the total number of true positive detections in the dataset.

6.3 CNN Performance

First, Figure 4 depicts the training and validation loss curves for the eight models that were trained as part of k-fold cross validation. Each model was trained on seven of the eight folds, and the eighth fold was used as a validation set. The training and validation loss curves demonstrate that the model was able to converge consistently, with variation that can be attributed to the stochasticity of the optimization scheme. Notably, the validation losses closely follow the training losses, suggesting that the models are able to generalize to unseen data that is from the same training distribution. This means that despite the small dataset size, the model did not overfit to the training data and was able to achieve similar losses during validation. Furthermore, the shaded region for both the training and validation curves, which bounds the performance of all eight folds, demonstrates that the lack of overfitting between training and validation splits was typical regardless of how the data was split.

In general, the models took less than an hour to train and had an inference speed of 4 ms on the NVIDIA GPU, highlighting the efficiency of this approach. Each trained model was then evaluated on the same, reserved

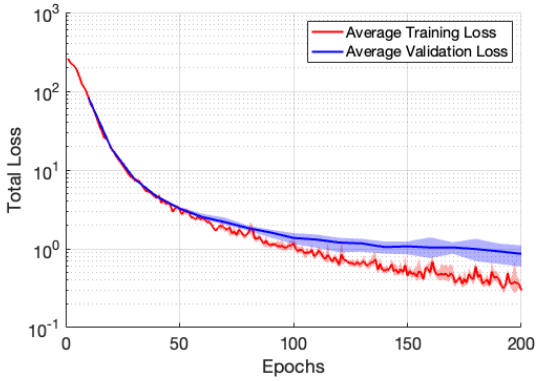

Figure 4: Training and validation loss curves shown on a logarithmic scale. The red and blue lines indicate the training and validation loss, respectively, averaged across all eight models. The training losses are reported as the average loss per epoch while the validation loss was computed every ten epochs. The shaded regions indicate the minimum and maximum values across all of the models trained.

Table 3: Performance of each of the eight folds used in k-fold cross validation.

Fold	Precision	Recall	L_2 Error [px]
1	0.939	0.909	0.586
2	0.926	0.952	0.556
3	0.927	0.933	0.605
4	0.930	0.894	0.580
5	0.909	0.933	0.608
6	0.922	0.931	0.586
7	0.940	0.918	0.577
8	0.908	0.920	0.570
Avg	0.925	0.923	0.584
Std	0.012	0.018	0.017

10 % test set that was not seen during training or validation. Table 3 demonstrates that the results for each fold were similar in performance across all three metrics. This consistent performance further demonstrates that the CNN is able to find underlying relationships in the data that are not unique to the specific subsets used for training each model.

Figure 5 shows three sample outputs from the CNN. The top two rows of images show examples of the CNN accurately capturing both the number and location of exoplanet signals, despite the difference in exozodi intensity. The bottom row, on the other hand, is a representative failure case, where the CNN overpredicts the number of signals. In particular, the CNN showed a tendency to overpredict in regions where two exoplanet signals are in close proximity. This is likely a limitation of the point source assumption that does not capture the PSF of the planets, such that the CNN erroneously predicts several point sources where in reality there are two extended sources.

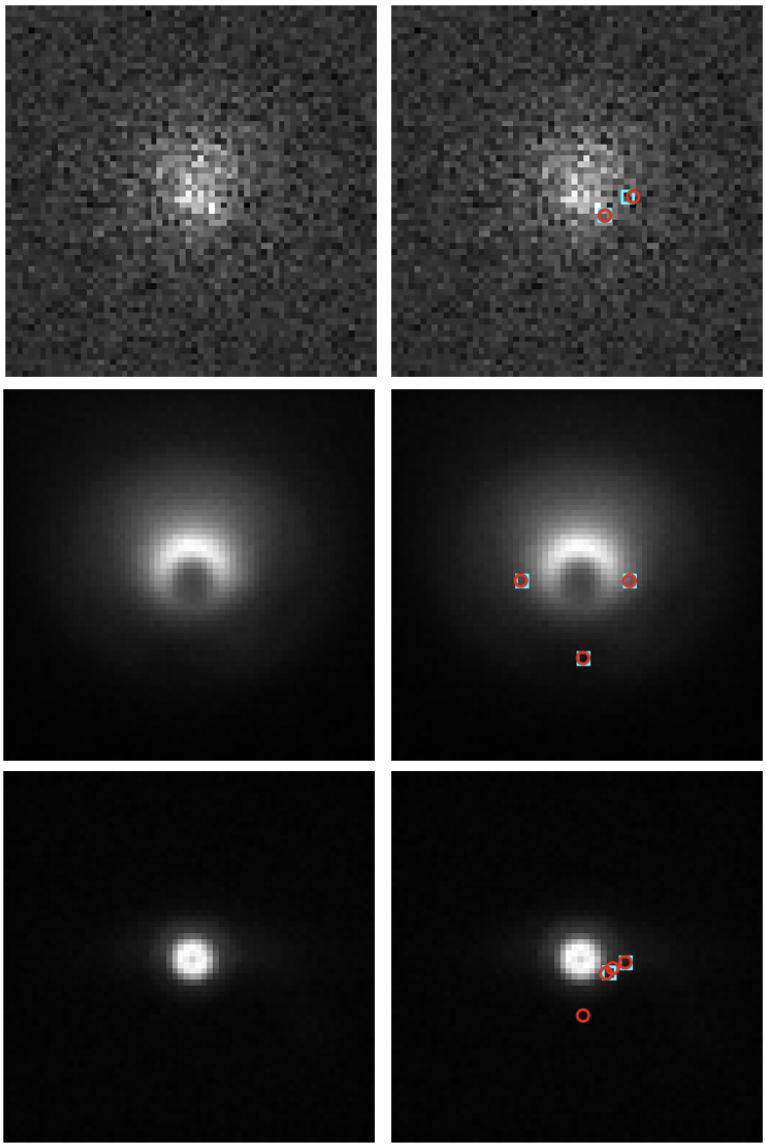

Figure 5: Sample outputs from the CNN developed in this work. The left column depicts the input image and the right column shows the input image annotated with the ground truth exoplanet signals (blue squares) along with the CNN predicted signals (red circles). All three images are simulated in the 615 - 800 nm passband and with degraded starshade performance (10^{-9}). (*Top*) This image simulates a σ Draconis system with a medium density resonant exzodi model and SNR = 5. (*Middle*) A τ Ceti system with high density resonant exozodi and SNR = 5. (*Bottom*) A ϵ Indi A system with low density resonant exozodi and SNR = 10.

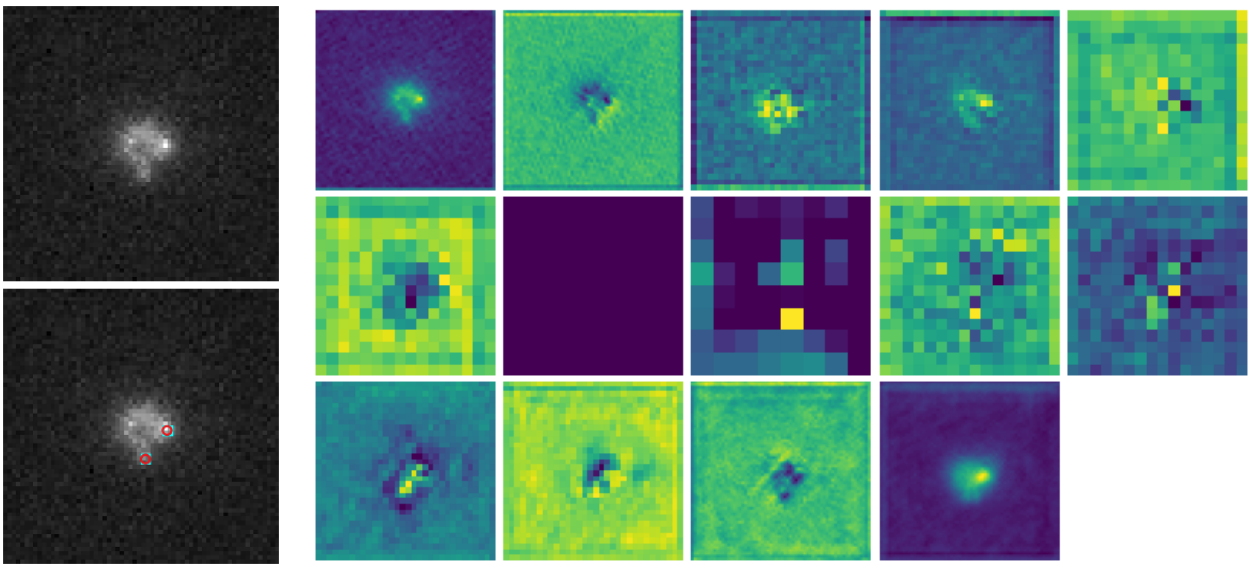

Figure 6: Feature maps for an input scene that models a σ Draconis system in the 425-552 nm passband with a low density, smooth exozodi model, SNR = 10 and nominal starshade contrast. (*Top Left*) Input scene. (*Bottom Left*) Input scene annotated with ground truth exoplanet signals (blue squares) and CNN predicted signals (red circles). (*Right*) A single feature map for each of the 14 convolutional layers in the CNN. The maps progress through the network along the rows, i.e. the first row includes the maps from the first five convolutional layers and so on.

6.4 Visualizing Feature Maps

The strength of CNNs lies in their ability to learn complex patterns through the use of non-linear activation functions and distributed representations of the input throughout the network. A common method of visualizing the inner representations of data is through the use of feature maps. Feature maps are the direct outputs of the convolutional layers of the network as an image is passed through, where each map is the result of a single filter operating on the input data. In the context of the U-Net architecture used in this work, the feature maps show how the encoder pathway progressively compresses the data into a high level representation. The decoder pathway then reconstructs the input back to its original spatial dimensions, ideally preserving the significant features necessary to detect the exoplanet signals.

Figures 6 and 7 show a single feature map for each of the 14 convolutional layers in the CNN architecture (Fig. 3). The first eight maps are from the encoding pathway and the final six maps are from the decoding pathway. Note that for the deeper layers in the network, each convolutional layer has 256 or 512 feature maps that vary depending on the learned weights for that filter. Therefore, each filter shown in Figs. 6 and 7 is meant to be a representative sample for its corresponding convolutional layer.

Specifically, Figure 6 shows the feature maps for an image from the test dataset where the CNN performed well. The maps for this image show that the two distinct exoplanet signals are preserved even in the most compressed representations. Figure 7, on the other hand, shows the maps for an image where the CNN performed poorly and overpredicted the number of exoplanet signals in a region where two exoplanet signals were in close proximity, similar to the bottom sample in Figure 5. Evaluating the feature maps for this sample, the distinct exoplanet signals are less clear, although the region where the two true signals are can be seen highlighted in the feature maps of the decoding pathways. This could indicate that the CNN is able to identify the exoplanet signals but does not have enough information to determine how to disentangle the number of signals.

Although it is difficult to identify exactly how the CNN makes its final predictions, visualizing the feature maps offers insights into how the CNN extracts and processes the input data and what information it might be encoding.

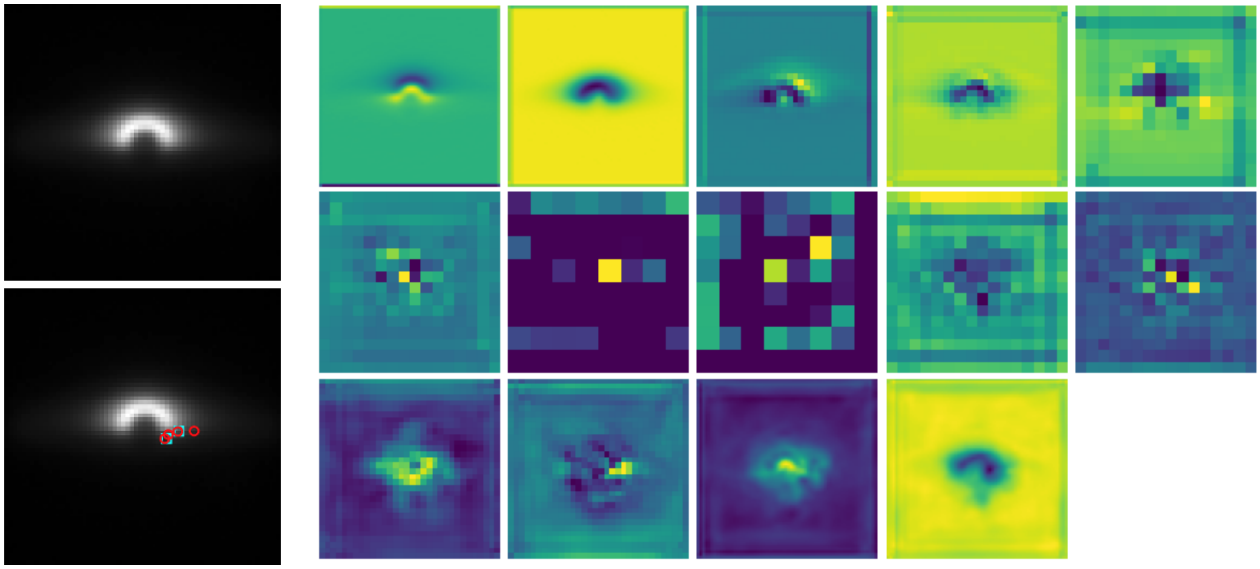

Figure 7: Feature maps for an input scene that models a ϵ Indi A system in the 615-800 nm passband with a medium density, resonant exozodi model, SNR = 10 and nominal starshade contrast. (*Top Left*) Input scene. (*Bottom Left*) Input scene annotated with ground truth exoplanet signals (blue squares) and CNN predicted signals (red circles). (*Right*) A single feature map for each of the 14 convolutional layers in the CNN. The maps progress through the network along the rows, i.e. the first row includes the maps from the first five convolutional layers and so on.

6.5 Comparison to State-of-the-Art

The results from the CNN were compared to two conventional approaches for exoplanet detection. Both of these approaches are characterized by background subtraction pipelines similar to those described in Section 2. The first approach investigates the application of SDI for starshade images by pairing the two broadband images for every scene and using the KLIP algorithm to estimate and remove the background noise. The second approach instead uses Independent Component Analysis (ICA) for background estimation and was developed specifically for SEDC. Therefore, these two approaches are complementary in that SDI is a well-established differential imaging technique for processing coronagraph images that has yet to be explored for starshades, whereas the ICA approach was specifically tailored for starshade images.

For the SDI approach, the pipeline first consists of pre-processing the data by pairing observations from each imaging band and separating the dataset by the exozodi model (resonant or smooth). Then, the paired images are passed through the KLIP algorithm using the pyKLIP implementation in SDI mode. The KLIP-subtracted images are then passed through a matched filter which is used to generate an SNR map.[19] Finally, the SNR map is used to compute the exoplanet detections. There was minimal tuning of parameters for this pipeline, which was implemented using existing functionality in pyKLIP. Additionally, it is important to note that SDI and the KLIP algorithm have been primarily studied for coronagraph images, and specifically used to calibrate background speckle noise that is not present in starshade images. Instead, starshade images are dominated by noise from sources such as solar glint, exozodiacal dust, and starshade formation flying errors, which are likely less correlated across wavelengths. Finally, SDI typically requires significantly more observations at narrower imaging bands in order to calibrate the data. For these reasons, the results obtained from SDI, shown in Table 4, can be interpreted as an initial comparison where the intent is to investigate the application of differential imaging techniques with limited imaging bands for starshades. These results show a low precision and recall when the pipeline was evaluated on the same test set and with the same definition of a true positive detection as used to evaluate the CNN. Additionally, these results show that the true positive detections are nearer to the 2 pixel detection criteria than the CNN results, although this is likely due to the fact that this pipeline does not have similar built-in point source assumptions as the CNN.

Table 4 also provides the precision and recall for the Quartus team from the Starshade Exoplanet Data Challenge. Their approach, which is documented in their final report on the SEDC website,[34] used ICA for background estimation. The ICA-subtracted images are then passed through a matched filter before local maxima detection, outlier detection including visual inspection, and finally thresholding to obtain the planet signal locations. The full results from the data challenge have not yet been released at the time of this paper, however, the initial analysis shows that results were able to achieve 60% precision overall, but were unable to detect approximately half of the true signals in the dataset when evaluated with the same criteria as the CNN and pyKLIP. The SEDC teams were not privy to the true locations of the exoplanet signals, which would have allowed them to further refine their approach on a portion of the dataset before evaluating it on the test dataset, similar to how the CNN was trained and evaluated. Even without ground truth information, the results from the ICA approach are significantly improved over the SDI results in all metrics, though a similar detection pipeline was used. This confirms that SDI with limited imaging bands is not well-suited for background estimation of starshade images and the highlights importance of visual inspection for accurate outlier detection. The SEDC team's final report specified that background subtraction was the limiting factor for disentangling the exoplanet signals, which further corroborates the conclusion from the SDI analysis. This indicates further promise for CNNs as an appropriate approach for exoplanet detection algorithms, as they do not require explicitly calibrating and subtracting the noise from the images.

Figure 8 shows a comparison of two images with the detections from the CNN versus the SDI pipeline and ICA detections. These examples show that SDI was unable to detect all exoplanet signals in the images and additionally made false positive predictions. However, as seen in the bottom row, the SDI approach typically successfully identified exoplanet signals with a higher SNR. This indicates that additional efforts to further tune the background estimation step and effectively increase the SNR of all exoplanet signals could lead to better results. Similar to the overall precision and recall trends shown in Table 4, the ICA results are significantly improved over the SDI results but not as accurate as the CNN results.

Overall, the results from these two approaches highlight the challenge of detecting and extracting planet information when the signals are embedded in exozodiacal dust, which tended to be a dominant source of noise in the SEDC images. Many of the existing methods for background estimation have been designed for ground-based observations using coronagraphs, where the largest source of noise is from speckles originating from the instrument and less so from the exozodi background.

6.6 CNN Generalization

The CNN's performance on the reserved test set are exceptionally promising, however as with any supervised machine learning application, it is important to test the limits of this performance to begin to quantify the robustness of the algorithm to various perturbations. Specifically, while CNNs have been shown to retain prediction accuracy as long as the input images in the test dataset belong to the same statistical distribution used during training, their performance often degrades when applied to images that are outside this training domain. The disparity between the in-distribution training domain and out-of-distribution test domain is often referred to as the domain gap, and is an active field of research[23–26] that is particularly relevant to this work in two ways. First, because the dataset used in this work is limited, the statistical distribution of images is also limited. For example, all 1,440 images simulate a total of 4 stars. Second, as with many spaceborne applications where it is difficult to obtain large, labeled datasets, there is a broader domain gap between the synthetic imagery used

Table 4: Comparison to conventional approaches. All results are reported for the same test set of 144 images with the same true positive criteria defined in Sec. 6.2.

Method	Precision	Recall	L_2 Error [px]
SDI	0.197	0.223	1.88
ICA	0.600	0.490	1.057
CNN	0.925 ± 0.012	0.923 ± 0.018	0.584 ± 0.017

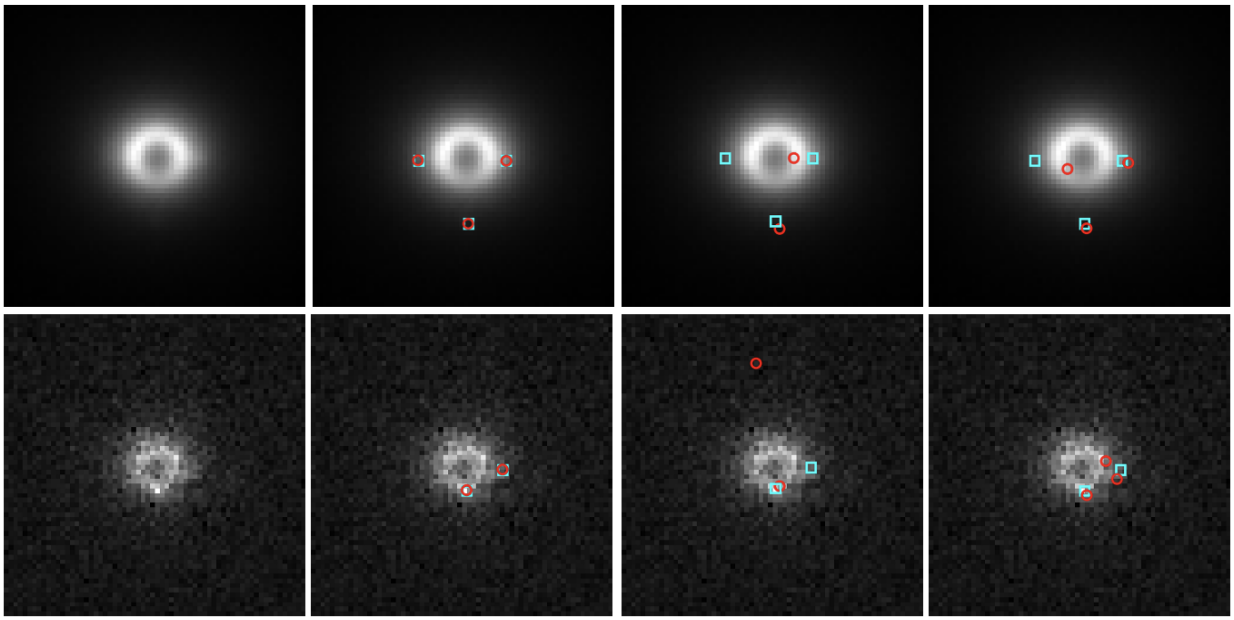

Figure 8: (*Top*) A τ Ceti system with medium density smooth exozodi, SNR = 20, and degraded starshade contrast (10^{-9}) simulated in the 615 - 800 nm passband. (*Bottom*) A σ Draconis system with high density smooth exozodi, SNR = 5, and nominal starshade contrast simulated in the 425 - 552 nm passband. (*Left*) Input scene. (*Middle-Left*) CNN predictions. (*Middle-Right*) SDI pipeline predictions. (*Right*) ICA pipeline predictions. The predictions are shown with red circles and the ground truth is shown with blue squares.

to develop machine learning algorithms and the real imagery that these algorithms will eventually be applied to. As a first step towards addressing the first domain gap, a handful of out-of-distribution images were generated using SISTER and passed through the CNN. The intention of this experiment was to qualitatively assess how well the CNN performed when the parameters of the input images varied from the range present in the SEDC dataset.

The results for two out-of-distribution image are shown in Figure 9. The two images simulate the star Formalhaunt, which was not included in the SEDC distribution, and are identical except for the number of exoplanet signals. The top image has zero exoplanet signals while the bottom has four exoplanet signals injected with $SNR \approx 6$. For both images, the CNN predicted the same false positive signal, indicating that it was not able to distinguish between the two images. In order to make quantitative assessments of the CNN's ability to generalize to out-of-distribution data, it is necessary to perform a much more extensive analysis that was not within the scope of this work. However, it is expected that at this stage, the CNN would not be robust to perturbations given the limited amount of training data. Instead, this initial qualitative assessment emphasizes the need for a dedicated effort to both create a larger and more diverse SISTER dataset for future iterations as well as improve the CNN architecture to incorporate physics models of exoplanetary systems, such that it can better generalize to out-of-distribution data.

7. SUMMARY AND FUTURE WORK

This paper introduces a novel approach for exoplanet detection using Convolutional Neural Networks (CNNs) and applies it to starshade images. Beginning with a semantic segmentation architecture, this work modifies the depth and output layer of the network to directly regress the locations of exoplanet signals. Then, the training loss is chosen to minimize errors in both the location and confidence of detected signals. The CNN results demonstrate high precision and recall when evaluated on a reserved test set, with inference speeds on the order of milliseconds. These results indicate that CNNs are a promising approach for accurate and efficient exoplanet image processing and warrant further development. Additionally, the approach developed in this work

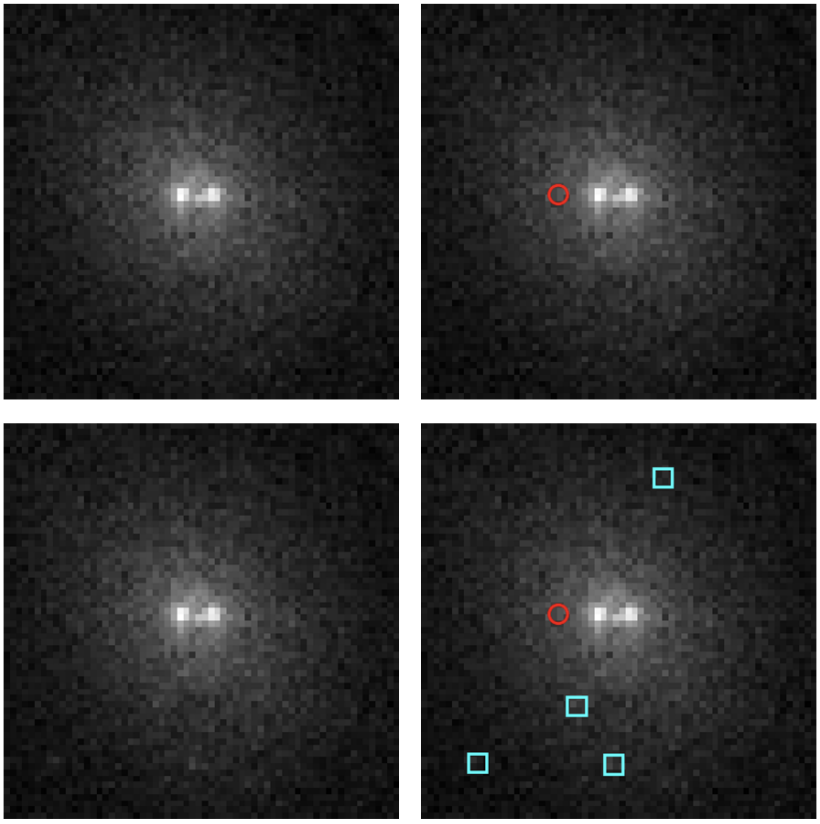

Figure 9: Two out-of-distribution images. The left column are the input images and the right column are the annotated predictions (red circles) and ground truth signals (blue squares). These images were generated using SISTER, derived from the example scenario 3 provided in the SISTER handbook.[35] The scene simulates the star Fomalhaut and was modified to include similar levels of exozodiacal dust and the same pixel pitch as the SEDC images. (*Top*) The scene without any planet signals. (*Bottom*) The scene with four exoplanet signals.

is not reliant on specific features of the starshade images, making it a versatile algorithm that can be applied to coronagraphs or other direct imaging payloads.

Moving forward, there are several opportunities to improve and extend this work. First, due to the limited amount of training data, the CNN indicated that it was not able to generalize when the host star and number of planets varied from the training distribution. Therefore, the primary focus of future work will be on expanding the starshade dataset to increase the robustness of the CNN and improve its ability to make inferences on out-of-distribution images. A comprehensive dataset is critical to thoroughly evaluate the performance of future supervised machine learning approaches and will additionally create opportunities for increased engagement between the machine learning and exoplanet science communities. Second, future work will also investigate ways to better tailor the CNN architecture for the task of exoplanet detection, such as incorporating physics models of the exoplanet signals to improve the CNN's ability to accurately identify several signals in close proximity. Finally, although the lack of explicit background subtraction was determined to be advantageous for the task of exoplanet detection, future work will investigate the incorporation of classical background estimation techniques to extend this work to exoplanet characterization. By explicitly separating the background noise from the exoplanet signals, the CNN can be used to extract the flux of the planet, which is critical for further characterization, including spectral analysis.

Image processing algorithms are critical to answering questions about the habitability of other worlds. As

direct imaging technology continues to mature, these algorithms must evolve to handle unprecedented amounts of data, while maintaining the performance necessary to uncover exoplanet signals among several sources of noise. Therefore, the CNN presented in this work is an important step towards enabling accurate, efficient, and versatile exoplanet image processing to achieve the ambitious direct imaging goals of the coming decades.

ACKNOWLEDGMENTS

The results in this paper have been derived using starshade simulations from the SISTER package. This work is enabled by the Starshade Technology Development Activity to TRL5 (S5), which led the Starshade Exoplanet Data Challenge (SEDC). The authors would like to thank Brian Dunne for agreeing to share his results from the SEDC that were used as a comparison in this work. This material is based on work supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE-1656518.

REFERENCES

[1] National Academies of Sciences, Engineering and Medicine, [*Pathways to Discovery in Astronomy and Astrophysics for the 2020s*], National Academies Press (2021).

[2] Tilley, D., Cleghorn, C. W., Thorat, K., and Deane, R., "Point Proposal Network: Accelerating Point Source Detection Through Deep Learning," in [*2021 IEEE Symposium Series on Computational Intelligence (SSCI)*], (2021).

[3] Park, T. H., Sharma, S., and D'Amico, S., "Towards robust learning-based pose estimation of noncooperative spacecraft," in [*2019 AAS/AIAA Astrodynamics Specialist Conference*], (2019).

[4] Ronneberger, O., Fischer, P., and Brox, T., "U-Net: Convolutional Networks for Biomedical Image Segmentation," (2015).

[5] Hildebrandt, S. R., Shaklan, S. B., Cady, E. J., and Turnbull, M. C., "Starshade Imaging Simulation Toolkit for Exoplanet Reconnaissance," *Journal of Astronomical Telescopes, Instruments, and Systems* **7** (2021).

[6] Cash, W., "Detection of Earth-like planets around nearby stars using a petal-shaped occulter," *Nature* **442**, 51–53 (2006).

[7] Vanderbei, R. J., Cady, E., and Kasdin, N. J., "Optimal occulter design for finding extrasolar planets," *The Astrophysical Journal* **665**, 794–798 (2007).

[8] D'Amico, S., Koenig, A., Macintosh, B., and Mall, S., "System Design of the Miniaturized Distributed Occulter/Telescope (mDOT) Science Mission," in [*33rd Annual AIAA/USU Conference on Small Satellites*], (2019).

[9] Seager, S., Kasdin, N. J., Booth, J., Greenhouse, M., Lisman, D., Macintosh, B., Shaklan, S., Vess, M., Warwick, S., Webb, D., D'Amico, S., Debes, J., Domagal-Goldman, S., Hildebrandt, S., Hu, R., Hughes, M., Kiessling, A., Lewis, N., Rhodes, J., Rizzo, M., Roberge, A., Robinson, T., Rogers, L., Savransky, D., Scharf, D., Stark, C., Turnbull, M., Romero-Wolf, A., Ziemer, J., Gray, A., Hughes, M., Agnes, G., Arenberg, J., Bradford, S. C., Fong, M., Gregory, J., Matousek, S., Murphy, J., Rhodes, J., Scharf, D., and Willems, P., "Starshade Rendezvous Probe Mission," in [*Bulletin of the American Astronomical Society*], **51**, 106 (2019).

[10] Seager, S., Turnbull, M., Sparks, W., Thomson, M., Shaklan, S. B., Roberge, A., Kuchner, M., Kasdin, N. J., Domagal-Goldman, S., Cash, W., Warfield, K., Lisman, D., Scharf, D., Webb, D., Trabert, R., Martin, S., Cady, E., and Heneghan, C., "The Exo-S probe class starshade mission," in [*Techniques and Instrumentation for Detection of Exoplanets VII*], *Proc. SPIE* **9605**, 273–290 (2015).

[11] Mather, J., Arenberg, J., D'Amico, S., Cash, W., Greenhouse, M., Harness, A., Hoerbelt, T., Kain, I., Kausch, W., Kimeswenger, S., Lisse, C., Martin, S., Noll, S., Peretz, E., Przybilla, N., Seager, S., Shaklan, S., Snellen, I., and Willems, P., "Orbiting Starshade: Observing Exoplanets at visible wavelengths with GMT, TMT, and ELT," *Bulletin of the American Astronomical Society* **51**, 48 (2019).

[12] Hu, M. M., Harness, A., Sun, H., and Kasdin, N. J., "Exoplanet detection in starshade images," *Journal of Astronomical Telescopes, Instruments, and Systems* **7** (2021).

[13] Marois, C., Lafrenière, D., Doyon, R., Macintosh, B., and Nadeau, D., "Angular Differential Imaging: A Powerful High-Contrast Imaging Technique," *The Astrophysical Journal* **641**, 556 (2006).

[14] Smith, W. H., "Spectral Differential Imaging Detection of Planets About Nearby Stars," *Publications of the Astronomical Society of the Pacific* **99**, 1344–1353 (1987).

[15] Lafreniere, D., Marois, C., Doyon, R., Nadeau, D., and Artigau, E., "A New Algorithm for Point-Spread Function Subtraction in High-Contrast Imaging: A Demonstration with Angular Differential Imaging," *The Astrophysical Journal* **660**, 770–780 (2007).

[16] Soummer, R., Pueyo, L., and Larkin, J., "Detection and Characterization of Exoplanets and Disks using Projections on Karhunen-Loeve Eigenimages," *The Astrophysical Journal* **755** (2012).

[17] Gonzalez, C. A. G., Absil, O., Absil, P.-A., Van Droogenbroeck, M., Mawet, D., and Surdej, J., "Low-rank plus sparse decomposition for exoplanet detection in direct-imaging ADI sequences. The LLSG algorithm," *Astronomy & Astrophysics* **589**, A54 (2016).

[18] Cantalloube, F., Mouillet, D., Mugnier, L. M., Milli, J., Absil, O., Gomez Gonzalez, C. A., Chauvin, G., Beuzit, J.-L., and Cornia, A., "Direct exoplanet detection and characterization using the ANDROMEDA method: Performance on VLT/NaCo data," *Astronomy & Astrophysics* **582**, A89 (2015).

[19] Ruffio, J.-B., Macintosh, B., Wang, J. J., Pueyo, L., Nielsen, E. L., De Rosa, R. J., Czekala, I., Marley, M. S., Arriaga, P., Bailey, V. P., Barman, T., Bulger, J., Chilcote, J., Cotten, T., Doyon, R., Duchêne, G., Fitzgerald, M. P., Follette, K. B., Gerard, B. L., Goodsell, S. J., Graham, J. R., Greenbaum, A. Z., Hibon, P., Hung, L.-W., Ingraham, P., Kalas, P., Konopacky, Q., Larkin, J. E., Maire, J., Marchis, F., Marois, C., Metchev, S., Millar-Blanchaer, M. A., Morzinski, K. M., Oppenheimer, R., Palmer, D., Patience, J., Perrin, M., Poyneer, L., Rajan, A., Rameau, J., Rantakyrö, F. T., Savransky, D., Schneider, A. C., Sivaramakrishnan, A., Song, I., Soummer, R., Thomas, S., Wallace, J. K., Ward-Duong, K., Wiktorowicz, S., and Wolff, S., "Improving and Assessing Planet Sensitivity of the GPI Exoplanet Survey with a Forward Model Matched Filter," *The Astrophysics Journal* **842**, 14 (2017).

[20] Wang, J. J., Ruffio, J.-B., De Rosa, R. J., Aguilar, J., Wolff, S. G., and Pueyo, L., "pyKLIP: PSF Subtraction for Exoplanets and Disks." Astrophysics Source Code Library, ascl:1506.001 (2015).

[21] Gomez Gonzalez, C. A., Absil, O., and Van Droogenbroeck, M., "Supervised detection of exoplanets in high-contrast imaging sequences," *Astronomy & Astrophysics* **613**, A71 (2018).

[22] Yip, K. H., Nikolaou, N., Coronica, P., Tsiaras, A., Edwards, B., Changeat, Q., Morvan, M., Biller, B., Hinkley, S., Salmond, J., Archer, M., Sumption, P., Choquet, E., Soummer, R., Pueyo, L., and Waldmann, I. P., "Pushing the limits of exoplanet discovery via direct imaging with deep learning," in [*Machine Learning and Knowledge Discovery in Databases*], 322–338, Springer International Publishing (2020).

[23] Park, T. H., Martens, M., Lecuyer, G., Izzo, D., and D'Amico, S., "SPEED: Next-generation dataset for spacecraft pose estimation across domain gap," in [*2022 IEEE Aerospace Conference (AERO)*], IEEE (mar 2022).

[24] Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., and Abbeel, P., "Domain randomization for transferring deep neural networks from simulation to the real world," in [*2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*], 23–30 (2017).

[25] Yang, J., Zhou, K., Li, Y., and Liu, Z., "Generalized out-of-distribution detection: A survey," (2022).

[26] Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J., "A theory of learning from different domains," *Machine Learning* **79**, 151–175 (2010).

[27] Hu, R., Hildebrandt, S. R., Damiano, M., Shaklan, S., Martin, S., and Lisman, D., "Starshade exoplanet data challenge," *Journal of Astronomical Telescopes, Instruments, and Systems* **7** (2021).

[28] Hildebrandt, S. R., Hu, R., and Damiano, M., "SEDC Reference Documentation - Broadband Imaging." https://exoplanets.nasa.gov/exep/technology/starshade-data-challenge (2021).

[29] Ribera, J., Guera, D., Chen, Y., and Delp, E. J., "Locating Objects Without Bounding Boxes," in [*2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*], 6472–6482 (2019).

[30] He, K., Zhang, X., Ren, S., and Sun, J., "Deep Residual Learning for Image Recognition," (2015). arXiv:1512.03385.

[31] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P., "Focal loss for dense object detection," (2020).

[32] Kingma, D. P. and Ba, J., "Adam: A Method for Stochastic Optimization," in [*3rd International Conference for Learning Representations*], (2015).

[33] Romero-Wolf, A., Bryden, G., Seager, S., Kasdin, N. J., Booth, J., Greenhouse, M., Lisman, D., Macintosh, B., Shaklan, S., Vess, M., Warwick, S., Webb, D., Ziemer, J., Gray, A., Hughes, M., Agnes, G., Arenberg, J. W., Bradford, S. C., Fong, M., Gregory, J., Matousek, S., Rhodes, J., Willems, P., D'Amico, S., Debes, J., Domagal-Goldman, S., Hildebrandt, S., Hu, R., Kiessling, A., Lewis, N., Rizzo, M., Roberge, A., Robinson, T., Rogers, L., Savransky, D., and Stark, C., "Starshade rendezvous: exoplanet sensitivity and observing strategy," *Journal of Astronomical Telescopes, Instruments, and Systems* **7**(2), 021210 (2021).

[34] Dunne, B., "Starshade Data Challenge Final Report." https://exoplanets.nasa.gov/exep/technology/starshade-data-challenge (2021).

[35] Hildebrandt, S. R. and Shaklan, S. B., "SISTER Handbook." http://sister.caltech.edu (2020).