LIGHTWEIGHT UNKNOWN SPACECRAFT TRACKING DURING PROXIMITY OPERATIONS

Kyle D. Shiller[1*] and Maruthi R. Akella[2]; [1]Graduate Research Assistant, [2]Professor, Cockrell Family Chair in Engineering #19 (The University of Texas at Austin, 2617 Wichita St, Austin, TX 78712).
[*][kyle.shiller@utexas.edu]

Abstract. *Increasing prevalence of spacecraft proximity operations necessitates highly accurate relative navigation solutions. However, many current applications are streamlined for operating about a fully known and coorperating spacecraft, which restricts generalized applications and reduces operational lifespan. This paper presents a novel Spacecraft Localization Pipeline, intended to detect and appropriately classify both known and previously unseen spacecraft within a monocular image. The pipeline is built around a convolutional neural network leveraging recent work in computer vision addressing unknown object detection. Simple metrics for associating spacecraft between images in a time-series are also proposed and analyzed, creating a complete pre-processing pipeline for higher level tasks.*

Introduction. Growing interest in space applications ranging from debris removal[1] to in-space servicing, assembly, and manufacturing (ISAM)[2,3] engenders the need for robust, autonomous spacecraft proximity operations. The problem of a non-cooperative target spacecraft – wherein the target provides no state information to the approaching spacecraft – drives the requirement for high-fidelity relative navigation solutions.

Due to restrictive size, weight, power, and cost (SWaP-C) constraints for small spacecraft platforms, many navigation suites employ monocular cameras coupled with computer vision algorithms. A growing body of work[4] employs convolutional neural networks (CNNs) to replace conventional image processing techniques in pose (position and orientation) estimation pipelines for relative navigation. CNNs exhibit more robustness to camera effects from the harsh space environment and have proven to be highly accurate object detectors. However, most of the available solutions are developed specifically for spacecraft known *a priori*, such that CNNs may be trained in advance using a known model. This severely limits operational life expectancy and restricts generalizability in the presence of previously unseen spacecraft or assets in a state of damage or dysfunction that require on-orbit repair services.

Aiming to bridge this gap, this work proposes the Spacecraft Localization Pipeline (SLP), depicted in Fig 1. The SLP is designed to ingest a time-series of monocular images and produce bounding boxes and classification labels for both known and unknown spacecraft in each frame, as well as associate each spacecraft instance from frame-to-frame. In the greater context of information fusion, specifically the Joint Directors of Laboratories (JDL) model,[5] the SLP serves the function of a Level 0 task, *source pre-processing*. The results from the SLP may thus be used in higher level tasks such as pose estimation or tracking. The architecture is designed to be lightweight for deployment on compute-limited, flight-grade hardware.

The SLP is comprised of two key components: the Spacecraft Localization Network (SLN) and the Spacecraft Association Module (SAM). The SLN is an object detection CNN designed for open world object detection and optimized for space application using several innovative loss terms. The focus of this work is the design and operation of the SLN. The SAM is a collection of two independent metrics devised to associate spacecraft between images. These metrics are presented to analyze feasibility as a final layer of processing on the SLN outputs in a complete pipeline.

At a high level, the SLP ingests a single monocular image, assumed to be part of a time series taken from one spacecraft approaching one or multiple other spacecraft. The SLN extracts known and unknown spacecraft from the image, classifies each appropriately, and returns appropriate bounding boxes and labels. The SAM then uses feature-based and spatial-based metrics to associate each detected spacecraft with those in previous images. The results may be output for use in higher-level tasks, as well as fed back into the SAM for association on the subsequent image.

The remainder of this paper is organized as follows. First, a literature review is provided for context, to show where the SLP fits into the broader context of spacecraft relative navigation. Then, the key design considerations of the SLP are covered in detail from both theoretical and practical standpoints. Results for each component of the design are presented with analysis, followed by closing remarks and suggestions for future work. Note that many of the results presented here are sourced from the primary author's thesis.[6]

Related Works. As a major motivation, a review of spacecraft pose estimation–specifically solutions employing monocular imagery – is first covered, building up to the role the SLP hopes to fulfill in this field. Coverage of open world object detection follows, as emerging work in that field lays the foundation for the SLN.

Spacecraft Pose Estimation. This work is largely inspired by the rapidly expanding field of spacecraft relative pose estimation. Due to the broad nature of this field, the following discussion is limited to pose estimation of an uncooperative spacecraft using a monocular camera. Earlier works in this area employed traditional computer vision techniques, ranging from Canny edge detection[7]

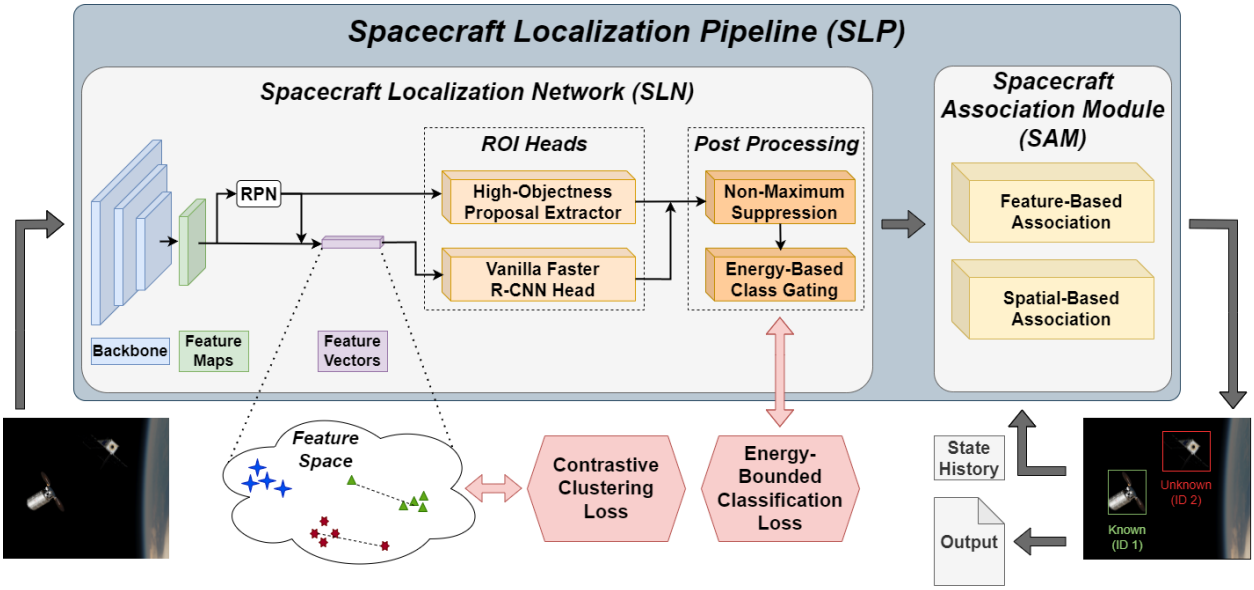


Figure 1. Spacecraft Localization Pipeline.

to the Sobel operator paired with the Hough transform.[8] These methods may be coupled with Perspective-n-Point (PnP)[9] algorithms to extract 3D locations of keypoints in the 2D image, thus recovering a pose solution. However, harsh lighting conditions and lens effects in space can significantly degrade these traditional solutions, leading to the rise of CNNs in spacecraft pose estimation.

CNNs were first employed for spacecraft pose estimation by classifying a spacecraft according to a discretized pose space.[10] This led to the Satellite Pose Estimation Challenge (SPEC2019),[11] in which all top competitors incorporated CNNs into their solution methods, either by direct pose regression[12] or a two-fold CNN approach.[13] Generally, the two-fold approach involved an object detection network coupled with a regression network, which regressed known keypoints for use in PnP or a similar algorithm to recover a static pose solution.

The two-fold approach matured in following years, leading to CNN-based pose estimation pipelines including filters for recursive estimation.[14] Tightly-coupled architectures regressed the known keypoints directly,[15] while loosely-coupled designs regressed the result from a static pose solver.[16,17] Later competitions, such as SPEC2021,[18] have seen continued growth in known spacecraft pose estimation pipelines. However, these methods all rely on known features of the target spacecraft, making *unknown* spacecraft pose estimation a distinct problem requiring different approaches.

Many proposed methods employ sensors beyond the capabilities of a monocular camera to leverage depth information to solve this problem. Some use the depth information available from stereo cameras,[19] where others use a Time-of-Flight (ToF) camera.[20] Others still move towards monocular-based solutions, but rely on single-beam LIDAR for depth measurements.[21,22] In an intriguing move towards using only monocular imagery, Park and D'Amico[23] proposed a novel CNN to approximate the 3D shape of an unknown spacecraft and simultaneously estimate pose.

Although this paper does not directly address pose estimation, the above discussion provides motivation for the SLP. Much of the unknown spacecraft pose estimation work above implictly assumes the spacecraft occupies a majority of a given image, so additional spacecraft or background noise are largely unaccounted for. The SLP is designed as a pre-processing pipeline to detect and classify each spacecraft in the field of view, such that each may be processed downstream by the appropriate higher-level algorithm.

Open World Object Detection. The design of the SLN rests on emerging work in CNNs dealing with Open World Object Detection (OWOD). A formal definition for OWOD was originally proposed by Joseph et al.[24] and builds on traditional object detection with the added objective of detecting and classifying previously unseen objects.

A number of solution methods employ CNNs with a class-agnostic region proposal network (RPN) to generate unknown proposals. Joseph et al. present an approach dubbed ORE,[24] which uses an energy function to augment classification and a novel clustering mechanism to enforce distinct grouping among classes. Liang et al.'s Unknown Sniffer (UnSniffer)[25] relies on their Generalized Object Confidence Score to improve unknown extraction, alongside an energy suppression scheme to better distinguish unknowns from background. Spatial Temporal Unknown Distillation (STUD)[26] aims to exploit the rich time and space information available in videos. STUD constructs "unknown counterparts" to each known class, which regularizes the decision boundary between known

and unknown classes.

Outside of RPN-based methods, an interesting method called RandBox[27] uses random proposals to negate the inherent bias of RPNs to favor object proposals that resemble known classes. After known object classification, a proposed matching score sweeps the remaining proposals for possible unknowns. Fang et al.'s MEPU framework leverages the diversity of foreground objects compared to standard backgrounds and characterizes unknowns based on the frequency of appearance in training. A secondary network is also trained using a proposed "localization quality" metric to extract additional unknowns. OW-DETR,[28] and extension on ORE, uses a transformer-based detector to make better use of the contextual information surrounding unknown objects. An attention-driven scheme enables single-stage detection, and thereby improved generalization. The pseudo-labeled unknowns are provided during training with an `unknown` label in conjunction with a "foreground objectness" branch to transfer knowledge of known classes, such as edge shapes, to the unknown classes.

These methods provide context for the wide ranging approaches to the OWOD problem that have emerged in the past five years. The SLN design is based on an OWOD approach for robustness and generalization considerations. Another possible approach could be creating a generic spacecraft detector coupled with an "unknown-aware" classifier to separate known and unknown spacecraft classes. However, in the interest of detecting a wider range of spacecraft than may be available for synthetic imagery, the SLN is instead built on an OWOD framework.

Spacecraft Localization Network. The SLN is a CNN-based computer vision algorithm intended to assign bounding boxes to any spacecraft in a provided monocular image. If a given spacecraft is known, it will be classified according to its known class; otherwise, the spacecraft will be labeled `unknown`.

The SLN architecture leverages a Faster R-CNN[29] baseline with a MobileNetV3[30] backbone. While other few-/single-shot detectors such as YOLOv8[31] are popular for mobile object detection, Faster R-CNN was selected largely for its Region Proposal Network (RPN). The RPN takes the feature maps output by the backbone to extract object proposals with associated "objectness" scores, which indicate the predicted likelihood that the proposal belongs to a known class. Next, pooling and fully connected layers compute a feature vector–a fixed-length, one-dimensional representation–corresponding to each proposal with an objectness score above a specified threshold. The feature vectors are then processed by two Region of Interest (RoI) heads–the regression head and the classification head–for prediction. These heads are fully connected layers that produce a bounding box and logits for each class (including the implicit background class) for a given proposal. This means a single proposal will yield $C+1$ (number of known spacecraft classes plus background class) bounding boxes with associated probabilities. The final output for each proposal is the box and probability corresponding to the highest scoring class.

Although a vanilla Faster R-CNN network generally provides a sufficient solution for known spacecraft detection, it fails to adequately address previously unseen spacecraft. This is largely due to overconfidence that the unseen spacecraft belongs to a known class or discarding the unseen spacecraft as background. However, several clever modifications enable the SLN to detect and appropriately classify unknown spacecraft.

These modifications are largely inspired by the open world object detector ORE.[24] The simple heuristic for pseudo-labeling unknowns is easily implementable and intuitive for space application, where images are assumed to contain limited unknowns (compared to terrestrial applications where many unknowns are present, e.g., a street view). The feature vector clustering is also appealing for distinguishing spacecraft from frame to frame. Several key design changes and additions were made to accommodate artifacts of operating in the complex space domain. Subsequent sections describe each component of the SLN in detail.

Unknown Proposal Extraction. Extracting unknown proposals is a crucial first step addressing the unknown spacecraft detection problem. This is where the RPN is a key enabler. By gathering all high-objectness proposals produced by the RPN and removing those that overlap known spacecraft, the remaining proposals form a set of potential unknown classes. These proposals are simply pseudo-labeled as unknowns for further evaluation, using the objectness scores as the probability that the proposal belongs to the `unknown` class. This task is accomplished by the intuitively-named High-Objectness Proposal Extractor, which works alongside the vanilla Faster R-CNN head. The results from each are concatenated and processed through non-maximum suppression (NMS), as depicted in Fig. 2.

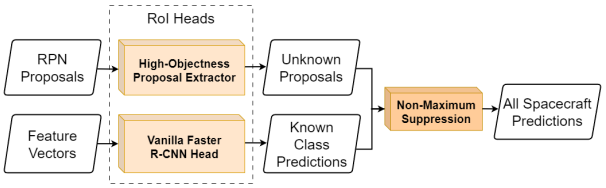


Figure 2. Unknown proposal extraction using high-objectness proposals.

NMS, depicted in Fig. 3, compares boxes that overlap by a specified margin. For boxes of the same class, NMS returns the highest-scoring box, as seen in the top example of Fig. 3. All unknown proposals are considered the same class for this process. For cases of mixed classes, as in the bottom example, NMS is customized to return the box with the highest known class score. This effectively already happens inside the vanilla Faster R-CNN head–here, the process merely "sweeps" the remaining proposals with low known-class predictions. In the event no

known classes exist for a set of overlapping boxes (i.e., only `unknown` predictions), the box with the highest objectness score is selected.

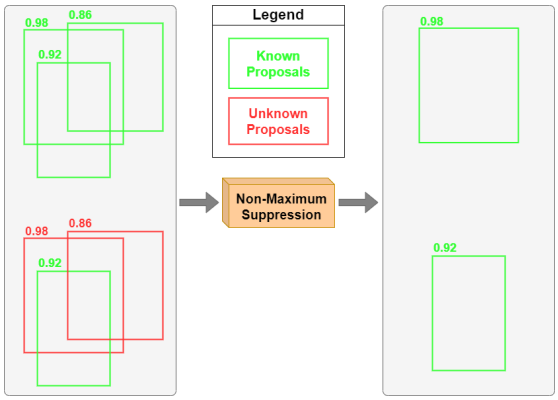


Figure 3. Non-maximum suppression with mixed known and unknown spacecraft.

It is worth noting that other networks use some form of objectness score as well. YOLO,[32] for example, uses a similar objectness metric using the overlap of some anchor and a known class. While this sounds strikingly similar to the RPN implementation, Dhamija et al.[33] show that YOLO performs quite poorly compared to Faster R-CNN on the open set detection problem. Open set detection is a close cousin of open world object detection which generally aims to address overconfident in-distribution predictions on out-of-distribution objects. YOLO's poor performance in this domain eliminated it as a candidate for the SLN architecture.

Energy-Bounded Classification Gating. The resulting known and unknown predictions are subsequently vetted through an Energy-Bounded Classification Gating. This procedure is intended to correct misclassified predictions using an *energy score*.

As was previously alluded to, vanilla Faster R-CNN often suffers from *overconfidence*, which generally refers to assigning known class labels to non-class objects. Sparing the theoretical analysis, it may be shown that all ReLU networks are prone to overconfidence issues.[34] ReLU networks are CNNs which use the rectified linear unit activation function, $\sigma(x) = \max\{0, x\}$, such as most object detectors.

Liu et al.[35] present several key insights to mitigate overconfidence based on energy scores. Leveraging early work in energy-based models[36] used for classification, it is first shown that the so-called *free energy* of a proposal $\mathbf{x}$ for a given classifier f is defined by the following:

$$E(\mathbf{x}; f) \triangleq -T \cdot log \sum_{i=0}^{C+1} e^{z_i/T} \tag{1}$$

where T is a temperature parameter, C is the number of known classes, and z_i is the class logits value for the i-th known class. Logits are the raw, unnormalized output of a classifier, and may be converted to standard probabilities using the softmax activation function.

This free energy score may be used to distinguish in-distribution (ID) data from out-of-distribution (OOD) data (i.e., known vs unknown spacecraft, respectively). In fact, Liu et al. show that the cross entropy loss used to train most classifiers actually *decreases the energy of ID data*. This gives rise to a simple ID-vs-OOD detection metric:

$$g(\mathbf{x}; \tau, f) = \begin{cases} 0 \text{ (OOD)} & \text{if} - E(\mathbf{x}; f) \leq \tau \\ 1 \text{ (ID)} & \text{if} - E(\mathbf{x}; f) > \tau \end{cases} \tag{2}$$

where τ is a tunable energy threshold and f represents the classification head. The negative energy, $-E(\mathbf{x}; f)$, is used to align with standard convention, where larger negative values indicate higher ID probability.

Liu et al. finally suggest a novel loss function to enforce greater separation between ID and OOD energy values. This Energy-Bounded Classification Loss is defined as follows:

$$\mathcal{L}_{energy} \triangleq \overbrace{\sum_{k \in \mathcal{K}} (\max(0, E(\mathbf{x}_k) - m_{id}))^2}^{\uparrow\text{ID proposal neg. energy}} + \underbrace{\sum_{k \notin \mathcal{K}} (\max(0, m_{ood} - E(\mathbf{x}_k)))^2}_{\downarrow\text{OOD proposal neg. energy}} \tag{3}$$

where m_{id} and m_{ood} define desired energy bounds and $\mathcal{K}$ is the set of known spacecraft classes. Recall, *negative* energy should be high for ID proposals and low for OOD proposals. The first term penalizes ID proposals with negative energy below the bound $-m_{id}$. Conversely, the second term penalizes OOD proposals with negative energy above the bound $-m_{ood}$. This may also be thought of as penalizing proposals with energy $E(\mathbf{x}) \in [m_{id}, m_{ood}]$.

Following these insights, Eq. 2 (with free energy as defined in Eq. 1) is employed as a drop-in OOD detector for the SLN. Notice that this requires no modifications to the model–this energy discriminator may be applied to any pre-trained classification model. However, in order to better leverage this capability, the energy-bounded loss is also applied during training to enforce greater distinction between known and unknown spacecraft.

Practically speaking, these additions enable a gating metric, which essentially vets the predicted labels from softmax scores using a judiciously chosen energy threshold τ. If the negative energy is below the chosen threshold for a predicted known spacecraft (ID), it may be relabeled as unknown (OOD).

Contrastive Clustering Loss. One final modification to the network lends itself back to ORE, the driving inspiration behind the SLN. A key component of ORE is a novel Contrastive Clustering loss, which encourages feature vectors of each known class to "cluster" together into

distinct groups in the latent space. The intent is to create a clear distinction between known classes by creating these clusters. This is depicted in Fig. 4.

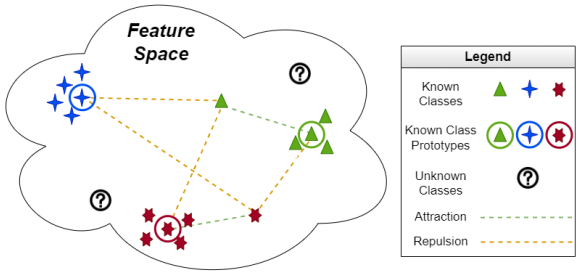


Figure 4. Latent space with Contrastive Clustering.

Contrastive clustering loss is formulated as follows:

$$\mathcal{L}_{\text{clstr}}(\mathbf{f}_c) \triangleq \sum_{i=1}^{C} \ell(\mathbf{f}_c, \mathbf{p}_i), \quad \text{where,} \tag{4}$$

$$\ell(\mathbf{f}_c, \mathbf{p}_i) \triangleq \begin{cases} \mathcal{D}(\mathbf{f}_c, \mathbf{p}_i) & i = c \\ \max\{0, \Delta - \mathcal{D}(\mathbf{f}_c, \mathbf{p}_i)\} & \text{otherwise} \end{cases}$$

where $\mathcal{L}_{\text{clstr}}$ is the total loss, $\mathbf{f}_c$ is a feature vector of known class c, $\mathbf{p}_i$ is a prototype feature vector of known class i, $\mathcal{D}$ is any distance function, and Δ defines some threshold for the acceptable distance between dissimilar objects. The ℓ term is posed as a *hinge embedded loss*, which penalizes two cases: (1) when objects of the same class are far apart, and (2) when objects of differing classes are closer than the established threshold Δ. The SLN implementation uses the simple Euclidean distance for the distance function, $\mathcal{D} := \|\mathbf{f}_c - \mathbf{p}_i\|_2$.

The prototype vectors are initially generated during training by averaging all feature vectors for a known class collected over a period of "burn-in" iterations. To account for maturing network weights, prototypes are periodically updated as a weighted average of the current prototype and a set number of stored feature vectors:

$$\mathbf{p}_c \leftarrow \xi \mathbf{p}_c + (1 - \xi) \frac{1}{n_c} \sum_{n=0}^{n_c} \mathbf{f}_{c,n} \tag{5}$$

This clustering is particularly useful for the proposed appearance-based association metric, which uses feature vectors to associate spacecraft between frames. Grouping feature vectors is intended to create greater distinction between each known spacecraft class.

Negative Samples & Hard Negative Mining. Two measures were also implemented at training time to further mitigate overconfidence issues: negative sample training and hard negative mining. Both methods are appealing as they may be implemented with any off-the-shelf object detector with no network modifications.

Negative sample training[37] aims to mitigate false positives due to background artifacts. Negative samples are simply images of distracting backgrounds without any ground truth labels. By supplying these images during training, the classification loss penalizes known class predictions on the negative samples, thus encouraging the network to ignore those backgrounds. In the spacecraft RPO context, negative samples generally include plain earth background. Earth images are readily available alongside existing synthetic training data, making negative sample training a natural addition to the overconfidence mitigation scheme. Some samples are shown in Fig. 5.

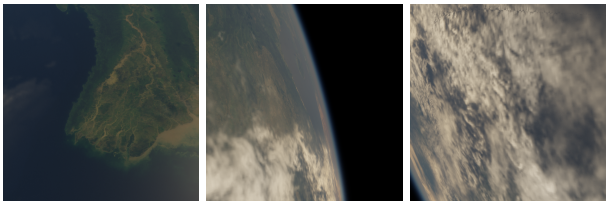
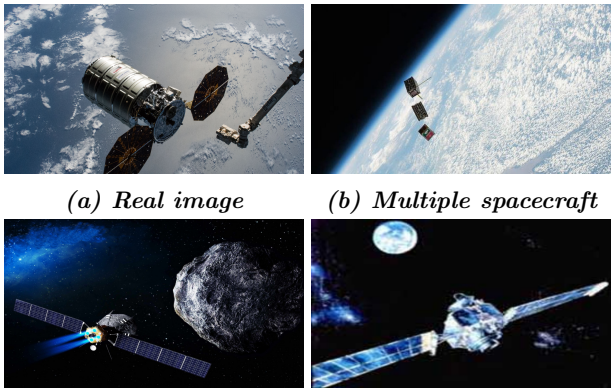


Figure 5. Example negative sample images.

Hard negative mining[38] takes this a step further by providing images with labels on particular distractors. Mining involves providing negative samples with ground truth data (bounding boxes and labels) specifying the hard negatives as null detections. Unfortunately, this labeled imagery is often not readily available.

In spacecraft detection, unknown spacecraft may be considered hard negatives, since they are likely to result in high-confidence in-distribution predictions. However, many publicly available datasets like URSO,[39] SPEED+,[40] and recently published SPEED-UE-Cube[41] do not inherently provide ground truth bounding box information, so some labeling process is still required. This is where the SatelliteDataset[42] provides utility. The dataset is drawn from a range of largely synthetic and some real images of spacecraft, all with ground truth bounding box data. Fig. 6 shows some sample images.



(a) Real image ***(b) Multiple spacecraft***

(c) Artist rendering ***(d) Unrealistic rendering***

Figure 6. Sample images from the SatelliteDataset.

It is important to note that many of the synthetic im-

ages are simple artist renderings, typically created for demonstration rather than realism. While many of these may seem unrealistic, for the purposes of the SLN, this is actually ideal. These negative samples help the network learn that spacecraft instances, or even merely instances of distinct, inorganic shapes and features, need not always belong to the known classes of spacecraft.

This raises the concern that this will train the detector to ignore previously unseen spacecraft. However, the RPN will still learn to extract the distinct spacecraft features. This process merely encourages the known spacecraft classifier to discard these proposals, which are later recovered during unknown pseudo-labeling. In fact, hard negative mining contributes greatly to contrastive clustering and energy-bounded training. By providing hard negatives during training, the network learns to penalize unknown spacecraft feature vectors that are near the known class clusters. Similarly, the energy-bounded loss penalizes energy values of hard negatives that are outside of the desired threshold.

Spacecraft Association Module. While the SLN detects spacecraft instances in individual frames, the task still remains to associate each spacecraft between frames. To this end, two metrics are proposed: a spatial-based metric and an appearance-based metric, collectively referred to as the SAM. The SAM effectively serves to refine measurements from the SLN before handling by higher level tasks.

Since the SLP is intended for near-range RPO approaches, these metrics largely rest on the assumption that spacecraft locations and attitudes will not change drastically between frames. While this may not be true for every image, it is a reasonable assumption over the course of an entire sequence of images. This two-fold approach is also designed to provide resilience against conditions that may compromise these assumptions.

During operations, the SAM would initialize spacecraft instances at their first appearance. In subsequent frames, spacecraft would be correlated to previous instances, regardless of classification label from the SLN, thus maintaining a sort of state history for each spacecraft. This feedback loop is seen in the lower right corner of Fig. 1.

As the scope of this work is focused on the development of the SLN, the SAM is not deployed in-the-loop. Rather, both association metrics are detailed and representative results are presented to display their efficacy.

Spatial-Based Metric. The spatial-based metric aims to correlate spacecraft instances based on bounding box locations. Again, assuming position will not change drastically between frames, this metric searches for relatively similar bounding boxes between each frame. This similarity is determined using Complete Intersection over Union (CIoU) loss.[43] Traditional IoU loss is defined as:

$$\mathcal{L}_{IoU} \triangleq 1 - IoU, \quad IoU = \frac{|A \cap B|}{|A \cup B|} \tag{6}$$

where A and B each represent a bounding box. IoU ranges between 0 and 1, where 1 indicates perfectly overlap and 0 indicates no overlap; therefore, greater overlap decreases IoU loss. CIoU loss extends this basic definition to account for the centroids and shapes of the bounding boxes as well. CIoU is written as follows:

$$\mathcal{L}_{CIoU} \triangleq 1 - IoU + \frac{d^2}{c^2} + \alpha\nu, \quad \text{where} \tag{7}$$

$$\nu = \frac{4}{\pi^2}(\arctan\frac{w_1}{h_1} - \arctan\frac{w_2}{h_2})^2, \quad \alpha = \frac{\nu}{(1 - IoU) + \nu}$$

The $\frac{d^2}{c^2}$ term penalizes separation by accounting for the difference between centroids, d, and the distance between the furthest corners, c, as depicted in Fig. 7. The $\alpha\nu$ term penalizes differences in aspect ratio, where ν compares consistency and α is a design parameter.

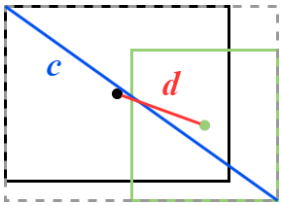


Figure 7. CIoU Loss visualization of displacement penalty.

Thus, under the previous assumptions, spacecraft may be correlated by selecting the candidate bounding box (from the previous frame) that minimizes CIoU loss.

Appearance-Based Metric. The appearance-based metric uses visual features, represented by the intermediate feature vectors from the SLN, to correlate spacecraft. In line with the assumptions above, spacecraft appearances should vary minimally from image to image, manifesting in similar feature vectors. Choosing the candidates that minimize the distances to each feature vector in the current image thus enables correlation by appearance. The Euclidean distance is used to mirror the contrastive clustering loss:

$$\mathcal{D}(\mathbf{f}_{\text{cand}_i}, \mathbf{f}_{\text{curr}_j}) \triangleq \|\mathbf{f}_{\text{cand}_i} - \mathbf{f}_{\text{curr}_j})\|_2 \tag{8}$$

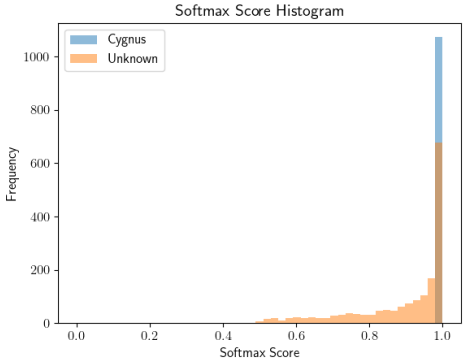
Contrastive clustering is particularly useful for this metric. Since feature vectors of known classes will be clustered in the latent space across a broad range of appearances, this distance metric offers a simple association criterion. This provides additional resilience in case the assumption that appearances will be similar is invalid for part of an image sequence.

Also, though unknown spacecraft are not clustered, hard negatives serve to increase the distance from the known classes. Coupled with the natural grouping of feature vectors among individual classes, and assisted by the assumption above, this still offers a simple heuristic for association between images.

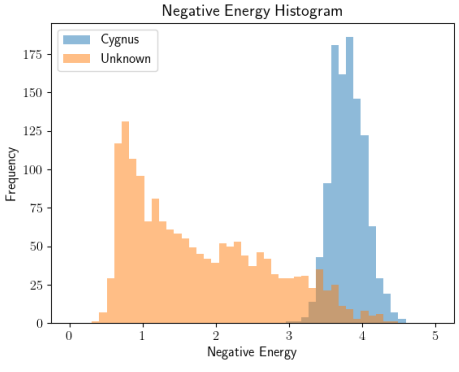
Results. Two networks were trained to compare results: a vanilla Faster R-CNN model and the SLN. Both are based on the torchvision implementation found in PyTorch.[44] Each was trained for 100 epochs using a step learning rate scheduler.

Since this work stems from previous pose estimation work by Kaki et al.,[16] the same Cygnus spacecraft was used as the single known class for testing. The recycled training dataset comprises over 8,000 synthetically generated images of Cygnus with various effects intended to represent those observed in real space imagery.[45] Augmentations from the Albumentations library[46] were also randomly applied with 0.2 probability for additional robustness in training.

For testing, a spread of SPEED+ Tango images and a sequence of SPEED-UE-Cube trajectory images are used. As these images do not contain ground truth bounding boxes, additional test images of Cygnus and Tango in random orientations as well as a simulated RPO trajectory were produced in-house and used to evaluate the SAM metrics.



(a) Softmax Probability Histogram

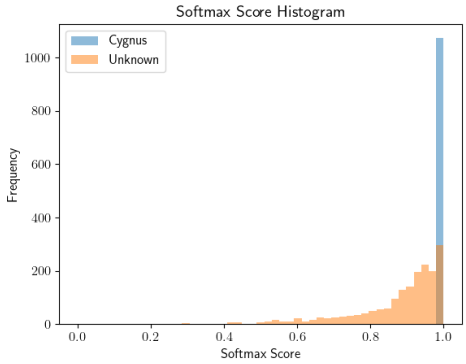


(b) Negative Energy Histogram

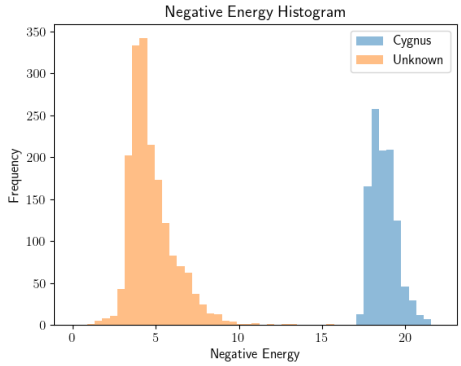
Figure 8. Before including energy-bounded loss.

Energy-Bounded Training Results. First, the impact of energy-bounded loss on softmax probabilities and energy scores is examined. Fig. 8 displays the results before including energy-bounded loss during training. Interestingly, the overconfidence inherent in ReLU networks manifests very clearly in the softmax probability scores. Almost all unknown spacecraft are classified as the known Cygnus class with confidence greater than 0.5, with a vast majority being incorrectly classified with greater than 0.9 confidence. A drop-in energy-bounded classifier proves much more effective at distinguishing the unknown spacecraft from Cygnus, without any changes to the model itself. However, there is still significant overlap that warrants further improvement.

Fig. 9 demonstrates the impact of incorporating energy-bounded loss. As intended, the unknown spacecraft is much more distinguishable from Cygnus by observing energy scores alone. The loss successfully pushes the scores for both the unknown and Cygnus classes beyond the desired bounds with minimal leakage. By setting a gating threshold between these bounds (e.g., energy score of 12), an energy-based gating mechanism would correctly discriminate almost all samples appropriately. This would satisfactorily mitigate misclassification using softmax probabilities, which are still largely overconfident for uknown spacecraft.



(a) Softmax Probability Histogram



(b) Negative Energy Histogram

Figure 9. After including energy-bounded loss.

Contrastive Clustering Results. To explore the impacts of contrastive clustering loss, the 1024-dimensional feature space is visualized in the 2-dimensional space using t-

distributed Stochastic Neighbor Embedding, or t-SNE.[47] This method essentially solves an optimization problem to minimize the difference between the joint probabilities of the high-dimensional data and a low-dimensional embedding.

Fig. 10 shows the t-SNE embedding of Cygnus feature vectors before and after implementing contrastive clustering loss. The tight grouping after including clustering loss, compared to the spread produced before, indicates the efficacy of clustering loss on a single class.

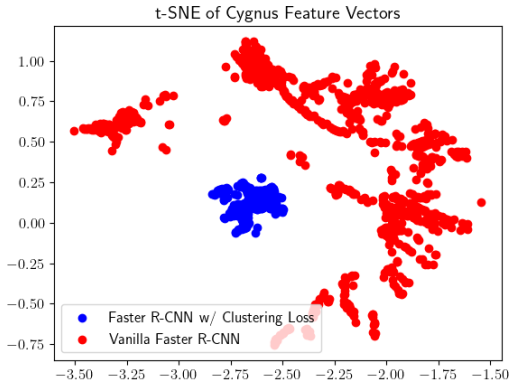


Figure 10. Comparison of Cygnus feature vectors before and after contrastive clustering loss.

Fig. 11 shows the t-SNE plot for the vanilla model exposed to Cygnus, Tango, and CubeSat images. While some inherent grouping occurs among each individual class, the groups are nearly co-located and overlap at the boundaries. This renders the SAM's spatial metric insufficient for distinguishing between spacecraft classes.

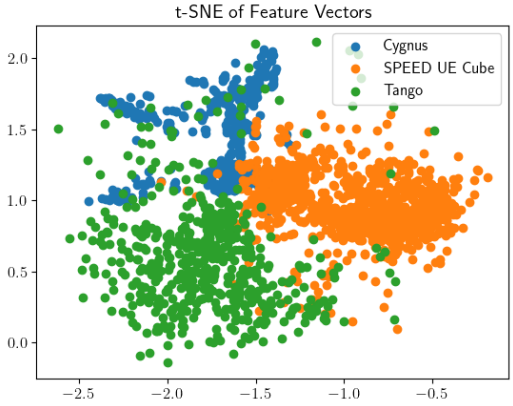


Figure 11. t-SNE plot of feature vectors before contrastive clustering loss (vanilla model).

Finally, Fig. 12 demonstrates the impact of incorporating contrastive clustering loss. The Cygnus feature vectors are clearly distinct from the bulk of the unknown predictions and form a much tighter cluster. Interestingly, the Cygnus cluster does not center around the prototype vector produced at the end of training, which is likely a function of the t-SNE solution. This indicates that the prototype vector may be effective when implementing clustering loss during training, but may not be well generalized outside of the training set. However, the neighborhood surrounding the prototype is still dominated by Cygnus feature vectors, which is encouraging for the SAM's appearance-based metric. Note that the unknown classes are still nearly co-located and more loosely grouped, as the specific unknown classes are unavailable for inclusion in clustering loss during training by definition.

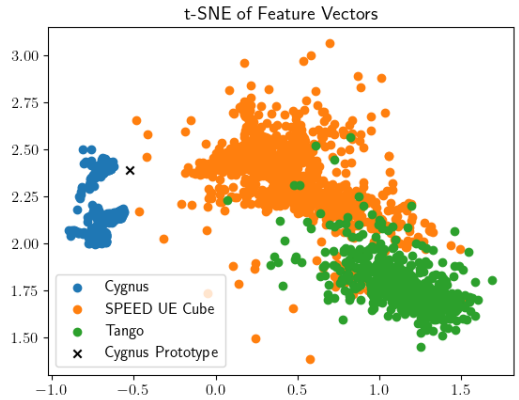


Figure 12. Feature vectors from SLN trained with contrastive clustering loss.

The influence of hard negatives in both the energy-bounded and contrastive clustering losses can not be understated. Without hard negatives serving as pseudo-unknown spacecraft, these loss terms only act on the known classes, with no impact on the unknown classes presented in testing. Consequently, the desired clustering and energy-bounded objectives are achieved, but unknown samples are largely mixed in with the known classes. This was observed during ablation testing when removing hard negatives from the training set. Even though hard negatives were incorporated to reduce overconfidence, they prove crucial to the success of these new loss terms.

SLN Performance. To evaluate the performance of the SLN, a few metrics must be introduced. Standard IoU, as introduced previously, is used to evaluate bounding box accuracy. To predict classification accuracy, the standard measures of precision, recall, and F1 score are used. These quantify true positives (TP), false positives (FP), and false negatives (FN) in various ways to highlight different aspects of performance. Precision, for example, quantifies the likelihood a detection belongs to the predicted class, and is useful when FPs are an important consideration. Recall quantifies the sensitivity to detect all spacecraft occurrences, and is useful when FNs are deemed important. F1 score is a harmonic mean of precision and recall, and is a useful blend of both metrics. Each is defined in Eqs. 9-11 below. Note that all three

metrics range between 0 and 1, with 1 being a perfect score.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{9}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{10}$$

$$\text{F1 Score} = \frac{TP}{TP + 0.5(FP + FN)} \tag{11}$$

Starting off, Table 1 shows the results of the vanilla model when evaluated on the simulated Cygnus and Tango RPO trajectory. To better differentiate performance, results are presented separately for each class as well as averaged to provide insight on general SLN performance. Despite impressive performance on the trained Cygnus class, the vanilla model possesses no inherent ability to detect unknowns, evident in the complete lack of Tango predictions. These results provide a baseline for Cygnus results to evaluate any degradation due to the changes that compose the SLN.

Table 1. Vanilla model performance on Cygnus and Tango RPO trajectory.

	Precision	**Recall**	**F1**	**IoU**
Cygnus	1.00	1.00	1.00	0.889
Tango	0.00	0.00	0.00	-
Weighted Avg.	-	-	-	-

Table 2 shows the SLN results when evaluated on the same trajectory images. Cygnus results only see minimal changes in the performance metrics, indicating no degradation in baseline performance due to the SLN modifications. The Tango results indicate that 93% of Tango predictions were actually Tango and that the SLN correctly detected 90% of Tango occurrences. IoU is remarkably worse for Tango compared to Cygnus, but this is to be expected. Unknown proposals–Tango, in this case–come from the RPN, which only learns to regress predicted bounding boxes based on known classes presented at training time. As a result, the IoU is anticipated to be lower for any unknown class compared to the known classes.

Table 2. SLN performance on Cygnus and Tango RPO trajectory.

	Precision	**Recall**	**F1**	**IoU**
Cygnus	1.00	1.00	1.00	0.906
Tango	0.93	0.90	0.91	0.525
Weighted Avg.	0.97	0.95	0.96	0.726

Fig. 13 shows representative SLN predictions on the trajectory test set. For these and future images, yellow boxes indicate Cygnus predictions, green boxes indicate the highest scoring unknown predictions, and red boxes indicate other unknown predictions. The top images showcase the SLN performing as desired. The bottom two images demonstrate typical failure cases of merged spacecraft predictions (left) and missed unknown detections (right), which result in the lower recall scores for Tango compared to Cygnus. Another miss case involved multiple unknown detections for a single instance of Tango, which contributes to the lower precision values.

Figure 13. Satisfactory (top) and degraded (bottom) SLN predictions on the RPO trajectory.

Next, Table 3 shows the results when the SLN is evaluated against randomized images of Cygnus and Tango during RPO. The results are similar to those for the RPO trajectory. The Cygnus IoU drops slightly, which is generally expected as the regression loss now contends with additional loss terms, so prediction may be slightly less optimized for bounding box regression. However, this may also be due to standard performance variation, as the Cygnus IoU is slightly improved in the RPO results.

Table 3. SLN performance on randomized Cygnus and Tango RPO images.

	Precision	**Recall**	**F1**	**IoU**
Cygnus	1.00	0.99	0.99	0.873
Tango	0.96	0.86	0.91	0.633
Weighted Avg.	0.98	0.93	0.95	0.762

Finally, Table 4 shows the classification statistics for the SLN tested on the SPEED+ and SPEED-UE-Cube images. IoU is not reported since ground truth boxes are unavailable for these test sets. As before, the SLN performs well on the Tango (SPEED+) images, with every instance of Tango being properly detected. However, performance is noticeably degraded on the CubeSat (SPEED-UE-Cube) images. All CubeSat instances are detected, but there are also a high number of FPs. Manual observation revealed that most FPs may be attributed

to background artifacts, as well as some duplicate predictions that are not appropriately filtered out.

Table 4. SLN performance on individual unknown spacecraft.

	Precision	Recall	F1	IoU
Tango	0.95	1.00	0.97	-
CubeSat	0.75	1.00	0.85	-

Figs. 14 and 15 show sample SLN predictions on assorted SPEED+ and SPEED-UE-Cube images, respectively. In each, the top images show successful predictions with generally accurate bounding boxes and appropriate classifications. False positives due to background artifacts (bottom-left of both figures) were the largest factor in the degraded performance metrics on these test sets. For Tango, spacecraft components in close range (Fig. 14 bottom-right) also generated false positives in some cases. Greatly oversized bounding boxes (Fig. 15 bottom-right) also appeared in both test sets. While these do not impact precision or recall, they could have notable impact on IoU measures. Overall, it is important to note that all detected spacecraft instances are correctly labeled as unknowns, with no FPs for Cygnus in these test sets.

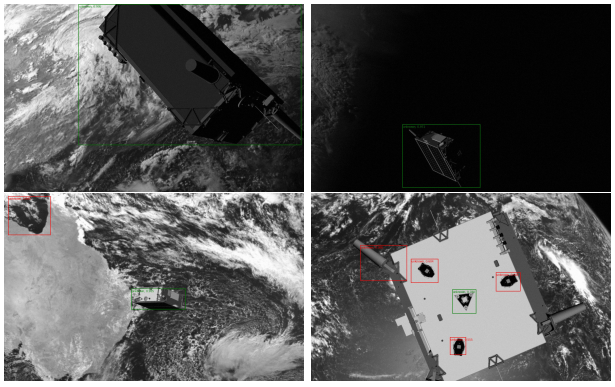


Figure 14. Satisfactory SLN predictions on the SPEED+ test set.

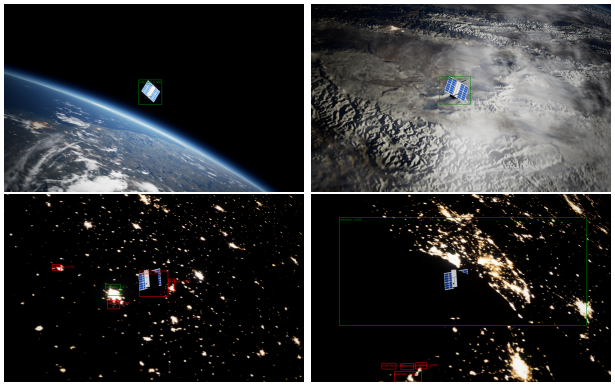


Figure 15. Satisfactory SLN predictions on the CubeSat test set.

SAM Performance. Although the SAM was not implemented in-the-loop for this work, the metrics were evaluated to determine the efficacy of the individual metrics. To accomplish this, each prediction in a given image is compared against the predictions in the previous image. This effectively creates a "comparison matrix" for each metric. For example, if three spacecraft are detected in sequential images, the comparison matrix will be three-by-three, comparing each possible pair of one current prediction with one previous prediction. Since the SAM computes loss and distance values, lower values indicate matches. Ideally, these should occur along the diagonal of the comparison matrices, indicating predictions are consistent between images.

Tables 5 and 6 show the averaged comparison matrices for the spatial- and appearance-based metrics, respectively, over the entire Cygnus and Tango RPO trajectory. Since there are only two spacecraft in this example, the matrices are two-by-two. Recognizing that Tango occasionally produces FPs, the corresponding values are nulled and ignored when averaging. As desired, the lowest values (highlighted in burnt orange) fall on the diagonals, indicating both metrics reliably match each spacecraft throughout the trajectory.

Table 5. Average of spatial-based comparison matrices over RPO trajectory.

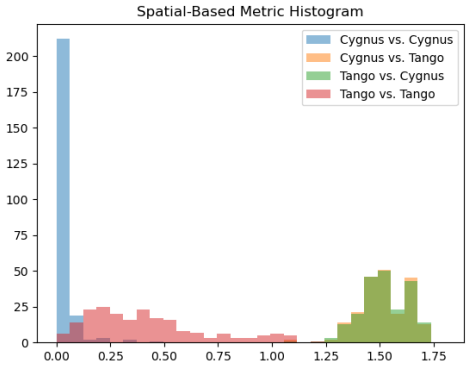
		Previous	
		Cygnus	Tango
Current	Cygnus	0.571	14.330
	Tango	14.332	2.391

Table 6. Average of appearance-based comparison matrices over RPO trajectory.

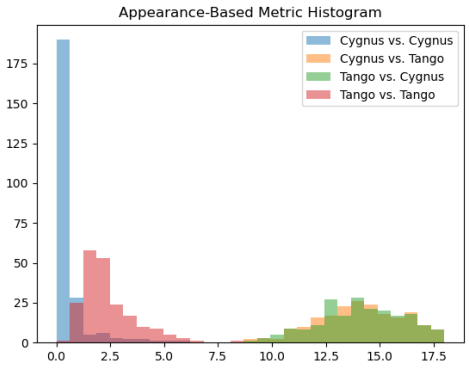
		Previous	
		Cygnus	Tango
Current	Cygnus	0.040	1.519
	Tango	1.520	0.417

Fig. 16 displays histograms for the values of each comparison matrix. Without exception, both metrics produce lower values for true matches compared to false matches (i.e., swapping Cygnus and Tango). Despite comparatively large variance for the Tango vs. Tango comparisons, the association metrics prove consistently reliable indicators of spacecraft association.

Conclusion. Spacecraft proximity operations continue gaining prevalence, resulting in an urgent call for robust relative navigation packages. This work proposes the Spacecraft Localization Pipeline for detecting and classifying known and unknown spacecraft classes during close-proximity navigation using only a monocular camera. The bulk of this work is the design of the Spacecraft Localization Network, which uses several novel loss terms to convert an RPN-based object detection network into an unknown-aware detector. Two novel metrics are



(a) Spatial-based metric



(b) Appearance-based metric

Figure 16. Histograms of association metrics during the RPO trajectory.

proposed, collectively referred to as the Spacecraft Association Module, to correlate spacecraft instances between frames, creating the full SLP.

Results validate the loss terms in accomplishing their desired training objectives and demonstrate promising results in detecting and classifying unknown spacecraft. Despite these encouraging results, several fail cases are prominent, often resulting from background artifacts. Results also indicate that both SAM metrics are particularly effective at associating spacecraft instances in a time-series.

Future work to address the SLN limitations may include incorporating the troubling background artifacts and more diverse spacecraft sizes during training. A refinement process, perhaps using traditional image processing techniques on an area of interest, may enable tighter bounding boxes on unknown proposals. Additionally, an incremental learning scheme may allow the SLN to gradually learn the unknown spacecraft classes. Beyond this work, implementing the SLP alongside higher level tasks–such as pose estimation or tracking–may test the efficacy of the SLP as a pre-processing pipeline. Deploying the SLP on lightweight, flight-grade hardware is another natural next step to verify readiness for operations.

Disclaimer. The views expressed in this paper are those of the authors and do not reflect the official policy or position of the United States Space Force, Department of Defense, or the U.S. Government.

References.

[1] C. P. Mark and S. Kamath, "Review of active space debris removal methods," *Space Policy*, vol. 47, pp. 194–206, 2019.

[2] N. T. Redd, "Bringing satellites back from the dead: Mission extension vehicles give defunct spacecraft a new lease on life," *IEEE Spectrum*, vol. 57, no. 8, pp. 6–7, 2020.

[3] D. Li, L. Zhong, W. Zhu, Z. Xu, Q. Tang, and W. Zhan, "A survey of space robotic technologies for on-orbit assembly," *Space: Science & Technology*, vol. 2022, 2022.

[4] L. Pasqualetto Cassinis, R. Fonod, and E. Gill, "Review of the robustness and applicability of monocular pose estimation systems for relative navigation with an uncooperative spacecraft," *Progress in Aerospace Sciences*, vol. 110, p. 100548, 2019.

[5] P. Lytrivis, G. Thomaidis, and A. Amditis, *Sensor Data Fusion in Automotive Applications*. 02 2009.

[6] K. D. Shiller, "Unknown spacecraft detection during proximity operations using neural networks," Master's thesis, The University of Texas at Austin, 2024.

[7] X. Du, Y. He, L. Chen, and S. Gao, "Pose estimation of large non-cooperative spacecraft based on extended pnp model," in *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 413–418, 2016.

[8] S. Sharma, J. Ventura, and S. D'Amico, "Robust model-based monocular pose initialization for noncooperative spacecraft rendezvous," *Journal of Spacecraft and Rockets*, vol. 55, no. 6, pp. 1414–1429, 2018.

[9] V. Lepetit, F. Moreno-Noguer, and P. Fua, "Epnp: An accurate o(n) solution to the pnp problem," *International journal of computer vision*, vol. 81, no. 2, 2009.

[10] S. Sharma, C. Beierle, and S. D'Amico, "Pose estimation for non-cooperative spacecraft rendezvous using convolutional neural networks," in *2018 IEEE Aerospace Conference*, pp. 1–12, 2018.

[11] M. Kisantal, S. Sharma, T. H. Park, D. Izzo, M. Märtens, and S. D'Amico, "Satellite pose estimation challenge: Dataset, competition design, and results," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 5, pp. 4083–4098, 2020.

[12] P. F. Proença and Y. Gao, "Deep learning for spacecraft pose estimation from photorealistic rendering," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6007–6013, 2020.

[13] B. Chen, J. Cao, A. Parra, and T.-J. Chin, "Satellite pose estimation with deep landmark regression and nonlinear pose refinement," in *ICCVW*, 2019.

[14] D. Kaidanovic, M. Piazza, M. Maestrini, P. Di Lizia, *et al.*, "Deep learning-based relative navigation about uncooperative space objects," in *INTERNATIONAL ASTRONAUTICAL CONGRESS: IAC PROCEEDINGS*, pp. 1–11, 2022.

[15] A. Harvard, V. Capuano, E. Y. Shao, and S.-J. Chung, *Spacecraft Pose Estimation from Monocular Images Using Neural Network Based Keypoints and Visibility Maps.*

[16] S. Kaki, J. Deutsch, K. Black, A. Cura-Portillo, B. A. Jones, and M. R. Akella, "Real-time image-based relative pose estimation and filtering for spacecraft applications," *Journal of Aerospace Information Systems*, vol. 20, no. 6, pp. 290–307, 2023.

[17] L. Pasqualetto Cassinis, R. Fonod, and E. Gill, "Review of the robustness and applicability of monocular pose estimation systems for relative navigation with an uncooperative spacecraft," *Progress in Aerospace Sciences*, vol. 110, p. 100548, 2019.

[18] T. H. Park, M. Märtens, M. Jawaid, Z. Wang, B. Chen, T.-J. Chin, D. Izzo, and S. D'Amico, "Satellite pose estimation competition 2021: Results and analyses," *Acta Astronautica*, vol. 204, pp. 640–665, 2023.

[19] V. Pesce, M. Lavagna, and R. Bevilacqua, "Stereovision-based pose and inertia estimation of unknown and uncooperative space objects," *Advances in Space Research*, vol. 59, no. 1, pp. 236–251, 2017.

[20] Z. Yan, H. Wang, L. Ze, Q. Ning, and Y. Lu, "A pose estimation method of space non-cooperative target based on orbfpfh slam," *Optik*, vol. 286, p. 171025, 2023.

[21] V. Capuano, K. Kim, A. Harvard, and S.-J. Chung, "Monocular-based pose determination of uncooperative space objects," *Acta Astronautica*, vol. 166, pp. 493–506, 2020.

[22] K. Matsuka, A. Santamaria-Navarro, V. Capuano, A. Harvard, A. Rahmani, and S.-J. Chung, "Collaborative pose estimation of an unknown target using multiple spacecraft," in *2021 IEEE Aerospace Conference (50100)*, pp. 1–11, 2021.

[23] T. H. Park and S. D'Amico, *Rapid Abstraction of Spacecraft 3D Structure from Single 2D Image*.

[24] K. J. Joseph, S. Khan, F. S. Khan, and V. N. Balasubramanian, "Towards open world object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5830–5840, jun 2021.

[25] W. Liang, F. Xue, Y. Liu, G. Zhong, and A. Ming, "Unknown sniffer for object detection: Don't turn a blind eye to unknown objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[26] X. Du, X. Wang, G. Gozum, and Y. Li, "Unknown-aware object detection: Learning what you don't know from videos in the wild," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[27] Y. Wang, Z. Yue, X. Hua, and H. Zhang, "Random boxes are open-world object detectors," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, (Los Alamitos, CA, USA), pp. 6210–6220, IEEE Computer Society, oct 2023.

[28] A. Gupta, S. Narayan, K. J. Joseph, S. Khan, F. S. Khan, and M. Shah, "Ow-detr: Open-world detection transformer," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9225–9234, 2022.

[29] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems* (C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, eds.), vol. 28, Curran Associates, Inc., 2015.

[30] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, "Searching for mobilenetv3," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1314–1324, 2019.

[31] Z. Huangfu and S. Li, "Lightweight you only look once v8: An upgraded you only look once v8 algorithm for small object identification in unmanned aerial vehicle images," *Applied Sciences*, vol. 13, no. 22, 2023.

[32] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Los Alamitos, CA, USA), pp. 779–788, IEEE Computer Society, jun 2016.

[33] A. R. Dhamija, M. Günther, J. Ventura, and T. E. Boult, "The overlooked elephant of object detection: Open set," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1010–1019, 2020.

[34] M. Hein, M. Andriushchenko, and J. Bitterwolf, "Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 41–50, 2019.

[35] W. Liu, X. Wang, J. Owens, and Y. Li, "Energy-based out-of-distribution detection," *Advances in Neural Information Processing Systems*, 2020.

[36] Y. LeCun, S. Chopra, R. Hadsell, A. Ranzato, and F. J. Huang, "A tutorial on energy-based learning," 2006.

[37] L. Gao, Y. He, X. Sun, X. Jia, and B. Zhang, "Incorporating negative sample training for ship detection based on deep learning," *Sensors*, vol. 19, no. 3, 2019.

[38] J. Li, C. Qu, and J. Shao, "Ship detection in sar images based on an improved faster r-cnn," in *2017 SAR in Big Data Era: Models, Methods and Applications (BIGSARDATA)*, pp. 1–6, 2017.

[39] P. F. Proença and Y. Gao, "Deep learning for spacecraft pose estimation from photorealistic rendering," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6007–6013, 2020.

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

[41] T. H. Park, Z. Ahmed, A. Bhattacharjee, R. Fazel-Rezai, R. Graves, O. Saarela, R. Teramoto, K. Vemulapalli, and S. D'Amico, "Spacecraft pose estimation dataset of a 3u cubesat using unreal engine (speed-ue-cube)," Jan 2024.

[42] H. Dung, B. Chen, and T. Chin, "A spacecraft dataset for detection, segmentation and parts recognition," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, (Los Alamitos, CA, USA), pp. 2012–2019, IEEE Computer Society, jun 2021.

[43] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-iou loss: Faster and better learning for bounding box regression," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 12993–13000, Apr. 2020.

[44] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32* (H. Wallach,

H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, eds.), pp. 8024–8035, Curran Associates, Inc., 2019.

[45] N. Dhamani, G. Martin, C. Schubert, P. Singh, N. Hatten, and M. R. Akella, "Applications of machine learning and monocular vision for autonomous on-orbit proximity operations," in *AIAA Scitech 2020 Forum*, p. 1376, 2020.

[46] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: Fast and flexible image augmentations," *Information*, vol. 11, no. 2, 2020.

[47] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008.