



## City Research Online

### City, University of London Institutional Repository

---

**Citation:** Song, J., Rondao, D. & Aouf, N. (2022). Deep learning-based spacecraft relative navigation methods: A survey. *Acta Astronautica*, 191, pp. 22-40. doi: 10.1016/j.actaastro.2021.10.025

This is the accepted version of the paper.

This version of the publication may differ from the final published version.

---

**Permanent repository link:** <https://openaccess.city.ac.uk/id/eprint/27112/>

**Link to published version:** <https://doi.org/10.1016/j.actaastro.2021.10.025>

**Copyright:** City Research Online aims to make research outputs of City, University of London available to a wider audience. Copyright and Moral Rights remain with the author(s) and/or copyright holders. URLs from City Research Online may be freely distributed and linked to.

**Reuse:** Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

---

---



# Deep Learning-based Spacecraft Relative Navigation Methods: A Survey

Jianing Song<sup>1,2,\*</sup>, Duarte Rondaio<sup>1,2</sup>, Nabil Aouf<sup>3</sup>

*City, University of London, ECV1 0HB London, United Kingdom*

---

## Abstract

Autonomous spacecraft relative navigation technology has been planned for and applied to many famous space missions. The development of on-board electronics systems has enabled the use of vision-based and LiDAR-based methods to achieve better performances. Meanwhile, deep learning has reached great success in different areas, especially in computer vision, which has also attracted the attention of space researchers. However, spacecraft navigation differs from ground tasks due to high reliability requirements but lack of large datasets. This survey aims to systematically investigate the current deep learning-based autonomous spacecraft relative navigation methods, focusing on concrete orbital applications such as spacecraft rendezvous and landing on small bodies or the Moon. The fundamental characteristics, primary motivations, and contributions of deep learning-based relative navigation algorithms are first summarised from three perspectives of spacecraft rendezvous, asteroid exploration, and terrain navigation. Furthermore, popular visual tracking benchmarks and their respective properties are compared and summarised. Finally, potential applications are discussed, along with expected impediments.

*Keywords:* Deep learning, Space relative navigation, Terrain navigation, Asteroid exploration

---

## List of Acronyms

		<b>3D</b>	three-dimensional
<b>2D</b>	two-dimensional	<b>AI</b>	Artificial Intelligence

---

\*Corresponding author

*Email addresses:* `jianing.song@city.ac.uk` (Jianing Song), `duarte.rondaio@city.ac.uk` (Duarte Rondaio), `nabil.aouf@city.ac.uk` (Nabil Aouf)

<sup>1</sup>Equal contribution

<sup>2</sup>Postdoctoral Research Fellow, Department of Electrical and Electronic Engineering

<sup>3</sup>Professor of Robotics and Autonomous Systems, Department of Electrical and Electronic Engineering

<b>ALHAT</b>	Autonomous Landing Hazard Avoidance Technology	<b>KPEC</b>	Kelvins Pose Estimation Challenge
<b>ANN</b>	Artificial Neural Network	<b>KRN</b>	Keypoint Regression Network
<b>BCE</b>	Binary Cross-Entropy	<b>LCLF</b>	Lunar-Centred, Lunar-Fixed Coordinates
<b>CD</b>	Crater Detection	<b>LoG</b>	Laplacian of Gaussian
<b>CI</b>	Crater Identification	<b>LRO</b>	Lunar Reconnaissance Orbiter
<b>CL</b>	Convolutional Layer	<b>LSTM</b>	Long Short-Term Memory
<b>CNN</b>	Convolutional Neural Network	<b>LVLH</b>	Local-Vertical, Local-Horizontal
<b>COCO</b>	Common Objects in Context	<b>ML</b>	Machine Learning
<b>CRO</b>	Candidate for a Regional Object	<b>MLP</b>	Multilayer Perceptron
<b>DEM</b>	Digital Elevation Map	<b>MSE</b>	Mean Square Error
<b>DL</b>	Deep Learning	<b>NASA</b>	National Aeronautics and Space Administration
<b>DMLP</b>	Deep Multi-Layer Perception	<b>NEA</b>	Near-Earth Asteroid
<b>DNN</b>	Deep Neural Network	<b>NN</b>	Neural Network
<b>DoF</b>	Degree-of-Freedom	<b>NST</b>	Neural Style Transfer
<b>DRCNN</b>	Deep Recurrent Convolutional Neural Network	<b>ODN</b>	Object Detection Network
<b>EKF</b>	Extended Kalman Filter	<b>PnP</b>	Perspective-n-Point
<b>ESA</b>	European Space Agency	<b>PDS</b>	Planetary Data System
<b>FCL</b>	Fully Connected Layer	<b>PyCDA</b>	Python Crater Detection Algorithm
<b>FPGA</b>	Field-Programmable Gate Array	<b>R-CNN</b>	Region-based Convolutional Neural Network
<b>GPOPS II</b>	General Purpose Optimal Control Software	<b>RANSAC</b>	Random Sample Consensus
<b>HDA</b>	Hazard Detection and Avoidance	<b>RGB</b>	Red-Green-Blue
<b>HRNet</b>	High-Resolution Net	<b>RMSE</b>	Root Mean Square Error
<b>ICP</b>	Iterative Closest Point	<b>RNN</b>	Recurrent Neural Network

<b>RoI</b>	Region of Interest	<b>SPN</b>	Spacecraft Pose Network
<b>RPN</b>	Region Proposal Network	<b>TRN</b>	Terrain Relative Navigation
<b>S/C</b>	spacecraft	<b>URSO</b>	Unreal Rendered Spacecraft On-Orbit
<b>SLAM</b>	Simultaneous Localisation and Mapping	<b>VO</b>	Visual Odometry
<b>SoC</b>	System-on-a-Chip	<b>WAC</b>	Wide Angle Camera
<b>SPEED</b>	Spacecraft Pose Estimation Dataset		

## 1. Introduction

In recent years, there has been a growing interest in Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL), especially amongst science, technology, engineering, and mathematics disciplines. There have been several approaches to define AI historically; the most common refers to techniques enabling machines to mimic human intelligence. Then, ML is the key component responsible for automatically processing data inside an AI. A Neural Network (NN) is a specific ML model aiming to approximate a certain function  $f^*$  relating training examples  $\mathbf{x}$  to labels  $y$  by defining a mapping  $y = f(\mathbf{x}, \boldsymbol{\theta})$  and learning the  $\boldsymbol{\theta}^*$  parameters that result in the best approximation. NNs work by stacking many different functions, called layers, and the number of layers defines the depth of the NN. The term DL derives from this wording, typically signifying a NN with large depth [1]. A rough relationship among these three concepts is summarised and illustrated in Fig. 1.

In the field of space exploration, autonomous vision-based spacecraft (S/C) navigation is one key area with the potential of greatly benefiting from DNN-based (Deep Neural Network) estimation methods. Cameras are rapidly becoming the preferred sensor for autonomous rendezvous thanks to the introduction of compact and lightweight passive optical sensors as feasible onboard instruments [2]. Additionally, vision-based techniques have been used in-flight for deep space navigation tasks [3]. Potential future applications of domains include: 1) non-cooperative rendezvous with a spacecraft; 2) terrain navigation for descent and landing; and 3) asteroid explorations and asteroid patch pinpoint localisation. All of these scenarios involve the estimation of a chaser or

---

<sup>4</sup><https://explore.mathworks.com/machine-learning-vs-deep-learning/chapter-1-129M-833I7.html>.

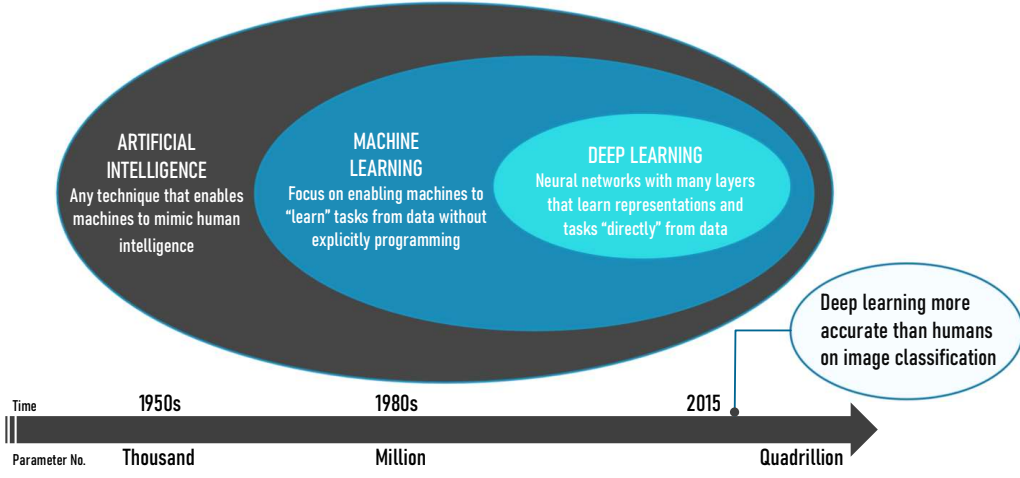


Figure 1: Relationship between AI, ML and DL (reproduced from Mathworks.<sup>4</sup>)

lander spacecraft's relative state, typically through the six Degree-of-Freedom (DoF) pose  $T_{ct}$  of the target object frame  $\mathcal{F}_t$  relative to the chaser frame  $\mathcal{F}_c$ , composed of a rotation  $R_{ct}$ , and a translation  ${}^c t_{ct}$  (see Fig. 2). Pose estimation methods have traditionally worked by relating features of the target (expressed in  $\mathcal{F}_t$ ), typically obtained from a model, to their images captured by the onboard camera (expressed in  $\mathcal{F}_c$ ), whereas using DNN models would adequately capture the intrinsic nonlinearities between the input sensor data and the state estimates, especially for images or Digital Elevation Maps (DEM).

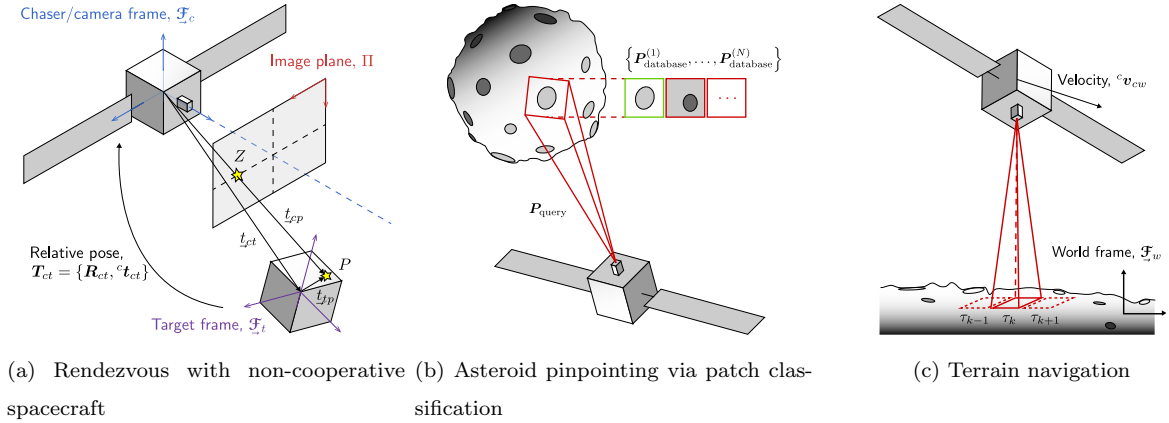


Figure 2: Identification of potential relative navigation scenarios for the application of DNNs.

Previous studies have approached the topic of DL-based navigation for space. Kothari et al. [4] collated various applications of DL for space, briefly discussing the achieved and prospective goals of

30 onboard systems for spacecraft positioning during docking and landing. Aiming at non-cooperative spacecraft rendezvous specifically, Cassinis et al. [5] first provided a review of CNN-based (Convolutional Neural Network) schemes in the context of monocular pose estimation systems discussing in detail several works (e.g. [6, 7, 8]). However, there is a shortage of comparative analysis of DL methods for general relative navigation in space. With this survey, we thus intend to bridge this

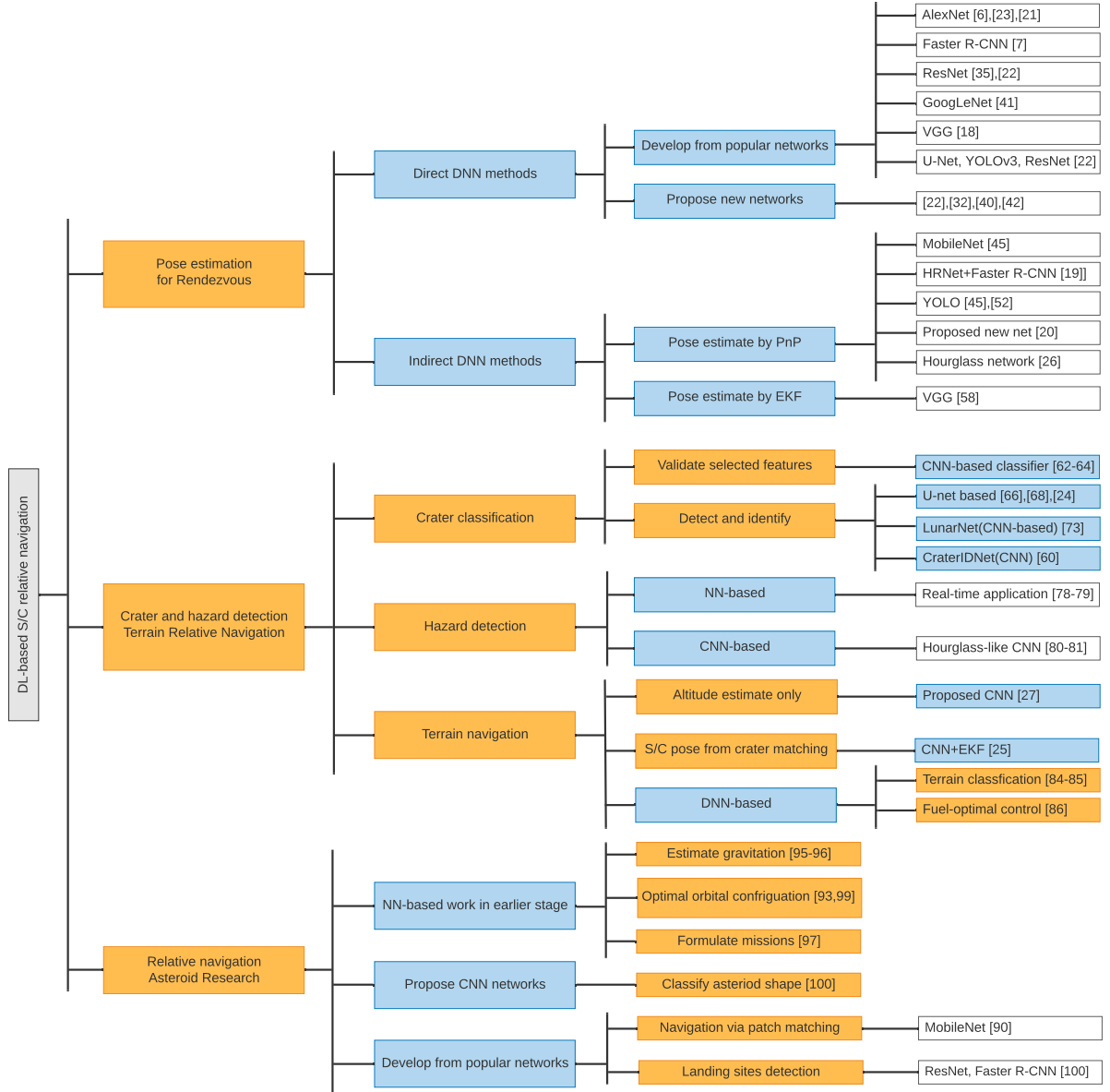


Figure 3: The tree diagram of DL-based S/C relative navigation approaches reviewed in this paper. The boxes in yellow, blue, and white represent applications, methods, and candidate references, respectively.

gap and provide a comprehensive reference for researchers and engineers aspiring to leverage deep learning for this subject, specifically for the three main applications identified in Fig. 2. Fig. 3 shows the set of research methods and application domains covered by our survey. In Fig. 3, direct DNN methods are end-to-end methods using DNNs, which constitute a direct, uninterrupted pipeline from inputs  $x$  to the desired quantity to estimate  $y$ . In contrast, indirect DNN methods are those in which the DNN is exclusively tasked with performing the image processing functions on the input, while the actual quantity to be estimated is achieved by combining this output with other methods, such as classical ML, geometry-based optimisation, and Kalman filtering.

This paper is organized as follows. Section 2 presents a review of DL-based pose estimation algorithms for spacecraft rendezvous. Section 3 contains a detailed review of crater and hazard detection of Terrain Relative Navigation (TRN) using DNNs. Section 4 provides a review of DL techniques with a focus on asteroid exploration. Finally, Section 5 lists the main conclusions and discussions.

## 2. DL-based Pose Estimation for Spacecraft Relative Navigation

### 2.1. Related Works on Terrestrial Pose Estimation

With the successful application of deep learning approaches in various research areas, DL-based camera-relative pose determination techniques for terrestrial scenarios have been attracting a considerable amount of interest.

Kendall et al. [9] proposed the PoseNet architecture for 6-DoF motion estimation in an end-to-end manner. To develop the pose regression network, they used a modified pre-trained GoogLeNet [10] by replacing all softmax classifiers with affine regressor. A weighted sum of the  $L^2$  error norms of the position vector and the attitude quaternion is selected as the loss function for better training of the location and orientation simultaneously. Their results demonstrate a 2 m and 3 deg accuracy for large scale outdoor scenes and 0.5 m and 5 deg accuracy indoors.

Rather than self-localising with respect to a known world model, Wang et al. [11] presented the DeepVO architecture to obtain a vehicle's egomotion from frame to frame based on monocular Visual Odometry (VO). The pipeline follows the architecture of a Deep Recurrent Convolutional Neural Network (DRCNN) [12], in which a pre-trained FlowNet [13] first learns features from sequences of Red-Green-Blue (RGB) images, which are then processed by Long Short-Term Memory



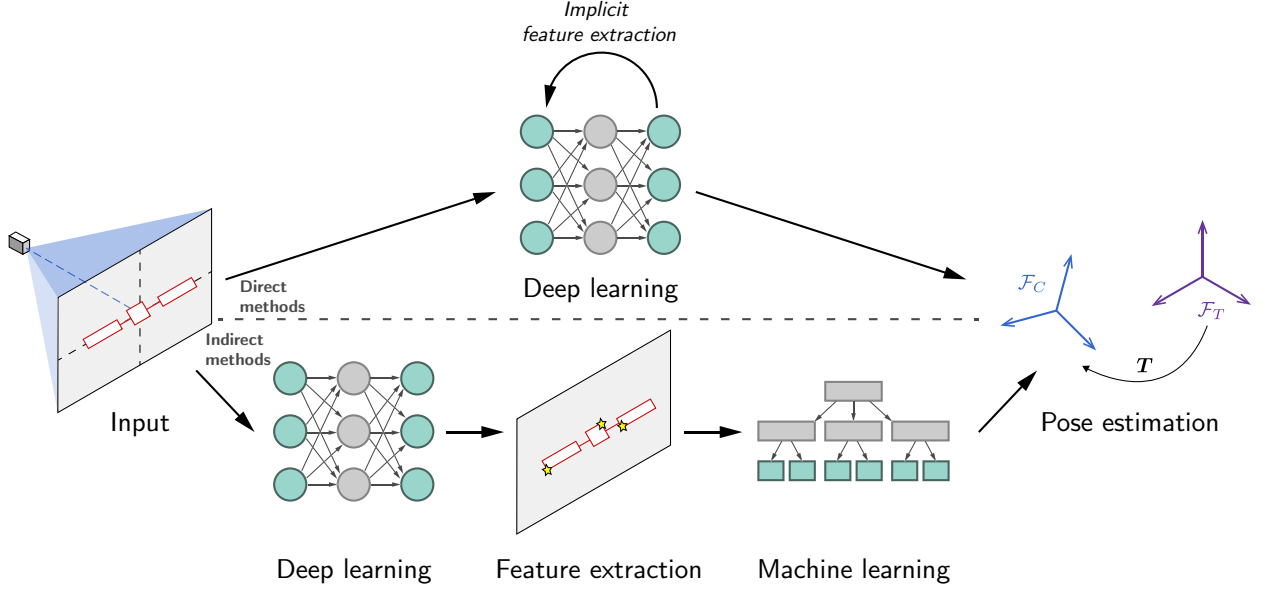


Figure 4: Direct versus indirect methods for DL-based pose estimation. The former use a DNN to directly estimate the pose from the input, whereas the latter use it exclusively to identify features, or landmarks, on the target, which are then input to a ML algorithm.

(LSTM) cells to estimate poses. The end-to-end DRCNN framework achieves an average Root Mean Square Error (RMSE) drift of 5.96 % and 6.12 deg per trajectory for position and attitude, respectively, on lengths of 100 m–800 m, showing a competitive performance relative to Monocular VISO2 [14].

Differing from the above end-to-end, or direct, methods, some works opt instead by following indirect methods, for which the DNN is exclusively tasked with performing the image processing functions on the input, while the actual quantity to be estimated is achieved by combining this output with other methods, such as classical ML or Kalman filtering. For instance, Rad and Lepetit [15] developed the BB8 algorithm for object pose estimation by combining a CNN to regress the two-dimensional (2D) locations of the eight three-dimensional (3D) points defining their bounding box with a Perspective-n-Point (PnP) algorithm [16] to retrieve the pose based on those correspondences. The VGG architecture [17] was chosen as the basis for their work, and the classical reprojection (or geometric) error was used as the corresponding loss function [18].

Figure 4 illustrates the difference between direct and indirect methods, which are explored further in this section.

## 2.2. Challenges and Motivations

Recent advancements in DL exhibit promising alternatives with respect to classical approaches, and related terrestrial frameworks also inspire the idea of DL-based spacecraft relative navigation. However, there still exists a gap between the two domains of application.

Relative pose estimation of objects in space is a different problem from pose determination of objects on Earth due to the vast differences in environment. Additionally, real labelled on-orbit images required for training DL algorithms are expensive and hard to obtain, which leads to a lack of space imagery datasets. Challenges in space missions for applying vision-based DL methods can be summarised from previous research [7, 19, 20, 21, 22, 23, 24, 25, 26] as follows:

- Planets and stars acting as background distractors for the navigation system;
- Challenging visual conditions due to lack of atmosphere and light diffusion;
- Much stronger shadows and varied illumination conditions resulting in extreme image contrast and low signal-to-noise ratio;
- Limited properties of space hardware in power consumption and computational resources (e.g., low sensor resolution);
- Training datasets for non-cooperative navigation of spaceborne objects are scarce;
- Concerns over the reliability of DL technique preventing their practice in the space industry.

The characteristics of space images also challenge conventional vision-based navigation algorithms for spacecraft, while the DL technique provides promising solutions and performance that can alleviate these issues. In terms of dynamic lighting, DNN-based schemes show increased robustness in attitude initialisation [22, 27]. With the deployment of high-performance devices, CNN-based methods can not only provide a lower computational complexity in pose acquisition process, but also reduce the need for complicated dynamics models [28]. Additionally, DNN pipelines are able to output various information and be combined with navigation filters or other processes [25].

Motivated by the attractiveness described above, and to overcome current limitations in spacecraft relative pose estimation, European Space Agency (ESA) launched the Kelvins Pose Estimation

Challenge ([KPEC](https://kelvins.esa.int/satellite-pose-estimation-challenge))<sup>5</sup> in 2019, inviting the community to propose and validate new approaches directly from greyscale images acquired by an on-board camera. The data for training and testing in this challenge consisted of Stanford’s Spacecraft Pose Estimation Dataset ([SPEED](#)), which contains labelled synthetic images of the Tango satellite, and a smaller, real set of images acquired in laboratory using a replica of the target.

### 2.3. Direct Frameworks for Spacecraft Relative Pose Estimation

In this survey, direct [DL](#)-based frameworks are defined as those in which the estimation of the desired quantity is entirely relayed to the [DNN](#), thus forming a continuous, uninterrupted pipeline from input to output. For spacecraft relative navigation, the problem is posited as estimating the 6-DoF pose of a target,  $\mathcal{F}_t$ , in the frame of reference of a chaser,  $\mathcal{F}_c$  (as shown in Fig. 2a).

The target may be non-cooperative, in which case it will not relay any explicit information to the chaser’s onboard navigation system, and the relative pose is estimated from acquired images of the target only.

By partitioning the relative pose space into discrete hypotheses, a classification framework may be established if the target spacecraft has a known model. Sharma et al. [7] have proposed a deep [CNN](#) for relative pose classification of non-cooperative spacecraft. Taking advantage of transfer learning, AlexNet model [29] pre-trained on the large ImageNet dataset [30] is modified by replacing the last few layers to adapt to the space imagery of the Tango spacecraft flown in the Prisma mission [31]. Ten datasets with different added noises are created from synthetic images. The proposed approach demonstrated greater accuracy than a baseline method using classical pose estimation techniques from [2D-3D](#) feature matching but is deemed not fine enough for any application other than a coarse initialisation.

Sharma and D’Amico [8] later on improve their original work with the creation of the Spacecraft Pose Network ([SPN](#)). The [SPN](#) (Fig. 5) uses a five-layer [CNN](#) backbone of which the activations are connected to three different branches. The first branch uses the Faster Region-based Convolutional Neural Network ([R-CNN](#)) architecture [32] to detect the [2D](#) bounding box of the target in the input image. To be robust towards intrusive background elements (i.e., presence of Earth), specific features output by the final activation map of the first branch are extracted using [R-CNN](#)’s Region

---

<sup>5</sup><https://kelvins.esa.int/satellite-pose-estimation-challenge>.

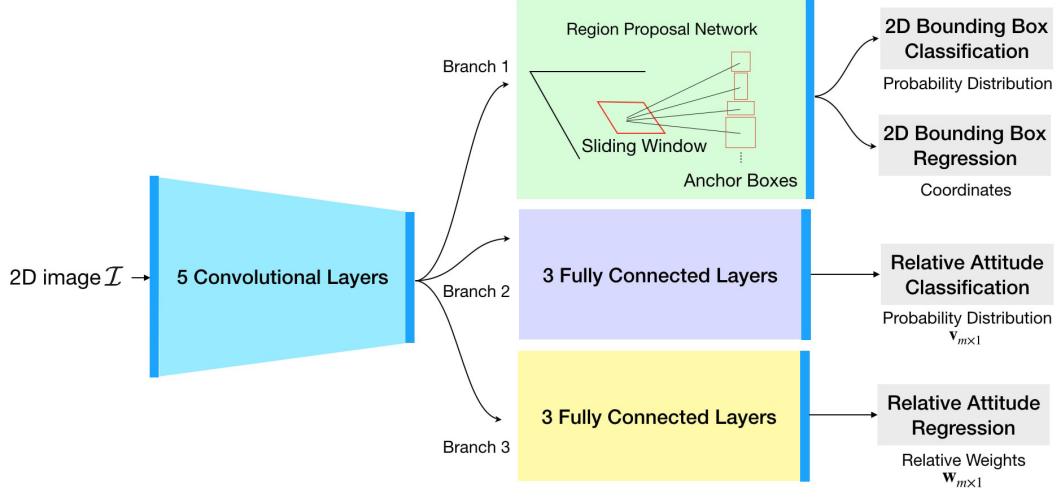


Figure 5: The Spacecraft Pose Network (SPN) architecture. Reproduced from Sharma and D’Amico [33].

of Interest (RoI) pooling technique, and then fed to the other two branches of the CNN containing three fully connected layers.

135 The second branch classifies the target attitude in terms of a probability distribution of discrete classes. It minimises a standard cross-entropy loss for the  $N$  closest attitude labels in the viewsphere. Lastly, the third branch takes the  $N$  candidates obtained from the previous branch and minimises another cross-entropy loss to yield the relative weighting of each. The final refined attitude is obtained via quaternion averaging with resort to the computed weights, which can be  
140 seen as a soft classification method.

Mathematically, the SPN utilises a Gauss-Newton algorithm to solve a minimisation problem for the estimate of relative position, for which the required initial guess is obtained from the bounding box (analogously to Kehl et al. [34]). The network is initially trained on the ImageNet dataset, and then the branch layers are further trained with an 80 %–20 % train-validation split on  
145 the SPEED dataset. As they report, the SPN method performs at degree-level and centimetre-level on relative attitude and position error, respectively.

In Ref. [33], Sharma and D’Amico expand their conference paper [8] by discussing two features of the SPN, target-in-target pose estimation and uncertainty quantification. The capability of estimating the uncertainty associated with the estimated pose of the SPN emphasises that SPN  
150 can be integrated with conventional navigation filters. Additionally, the authors detail the proposed SPEED dataset, considering the solar illumination of the synthetic images and the ground truth

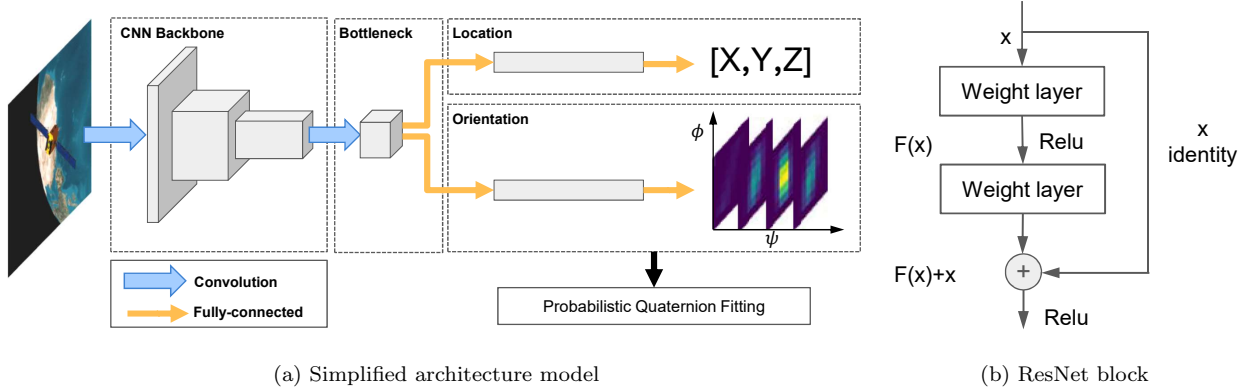


Figure 6: The CNN pipeline in Ref. [36]. The CNN front-end is based on ResNet, of which the elementary blocks implement skip connections that help mitigate the vanishing gradient problem in very deep networks.

calibration of the relative pose by the real images. The SPN is also trained in three versions by using different datasets, including SPEED, "Apogee Motor", "Imitation-25", and "PRISMA-25". Experiments are also carried out to demonstrate two key features of SPN method and compare it with their previous work, namely CNN-based [7] and image processing-based feature detection and correspondence [35] methods.

Instead of employing a bounding box feature detection, Proença and Gao [36] modify a pre-trained ResNet architecture [37] with initial weights trained on the Common Objects in Context (COCO) dataset to keep spatial feature resolution. Similarly to Ref. [8], two branches are designed to estimate 3D location and orientation, respectively. The position estimation consists of a simple regression branch with two fully connected layers and the relative error is minimised for better generalisation in terms of loss weight magnitudes. The continuous attitude estimation is then realised via a soft classification method [38]. Additionally, the authors present their own synthetic Unreal Rendered Spacecraft On-Orbit (URSO) dataset for training featuring Soyuz. Experiments on renders of URSO and SPEED datasets are conducted to evaluate the proposed framework, with which their model achieved a third and a second place on the synthetic and real test set categories of SPEED in KPEC, respectively. Moreover, the experimental results show that estimating the orientation by soft classification performs better than direct regression methods.

Hirano et al. [24] present a 3D keypoint estimator by using an AlexNet-based CNN architecture to regress spacecraft pose information directly, rather than retrieving 3D objects from the location of 2D keypoints. The parameters of AlexNet are changed for the purpose of the pose estimation

task, and batch normalisation layers [39] are utilised in all Convolutional Layer (CL) and Fully Connected Layer (FCL) for convergence in training. Synthesised images of a 3D model with the 3D keypoint position labels are generated on the Gazebo simulator [40] and used to train the CNN. Real images taken by hardware simulators are imported to evaluate the trained CNN. Images in both training and test dataset include the effects of lighting, shadows, and random noise, which leads the proposed framework to a potential application in practical space missions.

Arakawa et al. [41] also treat the attitude estimation as a CNN-based regression problem to obtain spacecraft attitude quaternion from the constructed images, in which the output of the proposed CNN is four independent real numbers corresponding to four quaternion elements. A 3D model of the JCSAT-3 satellite is built in the Blender software to generate a training image dataset. A point spread function is applied to the renders for simulating atmospheric fluctuations and optical effects. Compared with conventional image matching approaches, their results clarify an improved performance on the accuracy, robustness, and computational cost.

Considering that natural feature-based methods for spacecraft pose estimation are not always sufficient, Sonawani et al. [19] develop a modified model to assist a cooperative object tracker in space assembly tasks. The proposed CNN architecture is similar to Ref. [7], but uses VGG-19 as a backbone and replace the last layer with a 7-node one instead of an activation function. Two different models, namely a branch-based model and a parallel-based model, are developed to estimate relative poses. The frameworks of the two models are illustrated in Fig. 7, in which the parallel model contains two parallel streams for position prediction and attitude estimation, respectively. Synthetic images are generated in Gazebo, including truss-shaped objects labelled with the pose. The Euclidean distance error between the predicted poses and actual ones is defined as the loss function. Simulation results show their models are comparable to the current feature-selection methods and are robust to other types of spacecraft.

Aiming at vision-based uncooperative docking operations, Phisannupawong et al. [42] construct a spacecraft pose estimation model by proposing an advanced GoogLeNet pre-trained on URSO. The original GoogLeNet framework is modified by using 23 layers of CNN presented in Ref. [9], and the output for spacecraft pose is a seven-element vector. Experiments are carried out with an exponential loss function and a weighted Euclidean loss function, separately. The simulating results suggest that the weighted Euclidean-based pose estimation model successfully achieves moderately



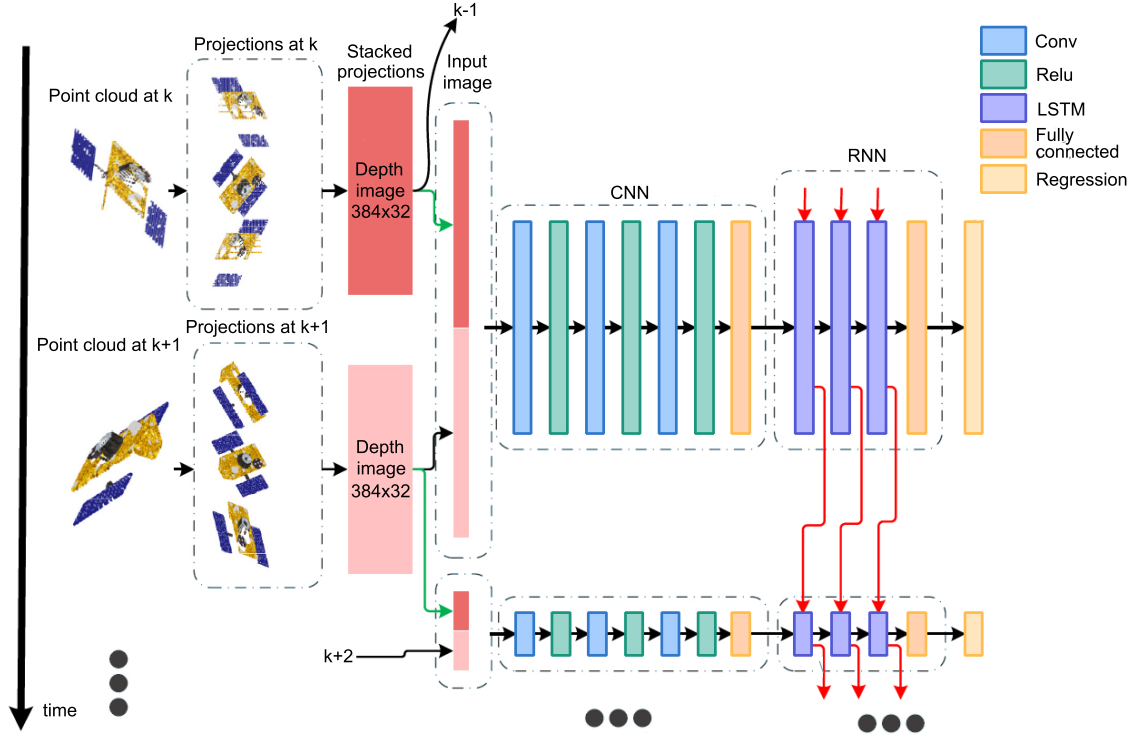


Figure 8: The DRCNN architecture of Kechagias-Stamatis et al. [43]. A shallow CNN architecture is utilised to extract low-level features for projected images, which are then modelled with LSTMs.

background on CNN performance. Thus, AlexNet, used as the backbone of the proposed framework, only changes the final FCL to yield attitude labels. The output attitude, obtained from a single branch unlike Ref. [8], is then refined using the eight most likely labels via direction cosine matrix averaging. Synthetic images of the SpaceX Dragon capsule are rendered using Blender at a fixed range of 20 m. Four different synthetic image sets are created to study the performance of the presented scheme and answer the proposed questions, namely considering a black and empty background, Sun angle variation, Earth background variation, and sensor noise. Based on their experimental results, it is indicated that: 1) both classification accuracy and attitude error exhibit an asymptotic trend; 2) the CNN performs well in more challenging light conditions of the Sun variation dataset but poorly for the Earth background and sensor noise datasets; and 3) using the confidence rejection threshold in the refinement step can improve estimation accuracy slightly.

Recently, Cosmas and Kenichi [23] first investigated the feasibility of CNN-based spacecraft pose estimation by assessing the onboard inference capabilities of the model. Accounting for power



Table 1: Summary of DL-based direct pose estimation methods for spacecraft relative navigation.

Ref.	Backbone	Transfer learning	Type	Dataset	Comments
[7]	AlexNet	ImageNet	Classifier	PRISMA, synthetic	Coarse initialisation
[8]	Faster R-CNN	ImageNet	Soft classifier	SPEED, synthetic	Introduction of SPN
[33]	SPN	ImageNet	Soft classifier	SPEED, PRISMA, synthetic (OpenGL)	Outperforms Ref. [7]
[36]	ResNet-50	COCO	Soft classifier	URSO (Soyuz S/C)	Soft classifier outperforms regressor in attitude
[24]	AlexNet	✗	Regressor	Synthetic (Gazebo), real	Direct 3D keypoint regression
[41]	2-layer CNN	✗	Regressor	Synthetic (Blender, JCSAT-3 S/C)	Evaluates robustness to noise, outputs quaternions
[19]	VGG-19	ImageNet	Regressor	Synthetic (Gazebo)	Cooperative object tracker
[42]	GoogLeNet	PoseNet	Regressor	Synthetic (Unreal Engine 4, Soyuz S/C)	Comparison of two loss functions
[43]	Shallow CNN + LSTM	✗	Regressor	Synthetic (Elite S/C), real (Envisat S/C)	Frame to frame motion estimator
[23]	U-Net, ResNet, YOLOv3	✗	Regressor	SPEED	Onboard FPGA implementation
[22]	AlexNet	ImageNet	Classifier	Synthetic (Dragon S/C)	Analysis of Sun angles, Earth presence, noise

consumption and cost-effectiveness, the Xilinx Zynq Ultrascale+ multiprocessor System-on-a-Chip (SoC) hybrid Field-Programmable Gate Array (FPGA) is proposed as a suitable solution. Two typical approaches and one presented framework are trained in Google Colab using the SPEED dataset, showing that a U-Net-based detection network performs better than the ResNet-50 based direct regression scheme, albeit poorer than the developed ResNet34-U-Net model. Later, the ResNet34-U-Net pipeline is implemented on the proposed hardware, starting with a YOLOv3 for RoI detection, followed by a landmark localisation network to predict keypoints. Inference experiments, including an evaluation of the performance, compared to a desktop-based implementation, DL processing unit resource utilisation, and power consumption are analysed with results of satisfactory accuracy and low on-chip power consumption of 3.5 W.

To make a clear comparison between the aforementioned approaches, Table 1 summarises the surveyed DL-based direct frameworks for relative pose estimation. As shown, over half of the solutions employ transfer learning, which traditionally has also been considered by the most successful applications of DNNs to terrain navigation problems. In terms of framework types, most are seen to adopt an estimation by regression or soft classifier. Open datasets of real images are limited; only PRISMA-25 and SPEED are available. On the other hand, synthetic imagery can be simulated by different software or platforms, such as OpenGL, Gazebo, Unreal Engine 4, and Blender, leading to datasets such as URSO. Moreover, the recent research output volume demonstrates there is an increasing interest in DL-based spacecraft pose estimation, including even a first report on onboard implementations with FPGAs.

#### 2.4. Indirect Frameworks for Spacecraft Relative Pose Estimation

Estimating the pose from images using end-to-end DL-based methods has been argued to yield inadequate feature representation and limited explainability, either of which has so far achieved subpar performances as opposed to geometry-based methods. Sattler et al. [45] discuss the limitations of end-to-end CNN-based terrain pose regression and suggest that there is a gap for practical applications. Moreover, the DNN model has a risk of overfitting, which results in unpredictable drops in performance between the training images and test images due to memorising, rather than learning, properties of the former set that do not function well on the latter [1]. Therefore, some research avenues have recently refocused on the indirect methods, which aim to combine DL and conventional geometry-based techniques to refine the estimation of the pose.

To promote the practical use of DL-based pose estimation in space missions, Park et al. [46] take the SPN framework [8] and modify it by employing both a novel CNN for target detection and Random Sample Consensus (RANSAC) algorithms for solving the PnP problem. The proposed CNN is decoupled into the detection and pose estimation networks to determine the 2D bounding box of the RoI and to regress the 2D locataion of keypoints, respectively. As demonstrated in Fig. 9a, the Object Detection Network (ODN) and the Keypoint Regression Network (KRN) closely follow the pipeline of YOLOv2/YOLOv3 [47], but use MobileNetv2 [48] and MobileNet [49], respectively. To drastically reduce the number of network parameters, traditional convolution operations of the network are replaced with depth-wise convolutions followed by point-wise convolutions (see Fig. 9b).

Considering the lack of real space-based datasets with representative texture and surface illumination properties, Park et al. [46] also contribute with a new training procedure to improve the robustness of CNNs to spaceborne imagery when trained solely on synthetic data. Inspired by Ref. [50], they generate a new dataset by applying neural style transfer techniques [51] to a custom synthetic dataset with the same pose distribution as SPEED. After training with the new texture-randomised dataset, the proposed network performs better on spaceborne images and scores 4th place in KPEC.

The 1st place KPEC solution is also an indirect DL-based scheme proposed by Chen et al. [20], where DL and geometric optimisation are combined to present a CNN-based pipeline for pose estimation from a single image. Firstly, 3D landmarks of the satellite are computed from

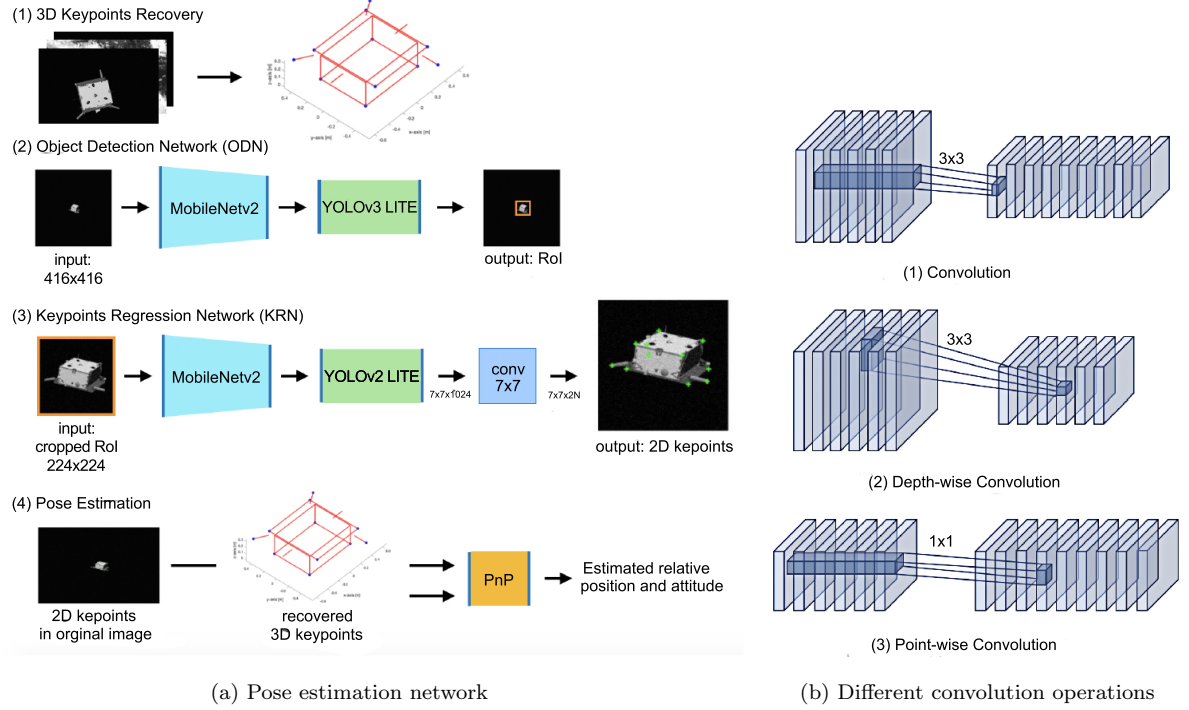


Figure 9: Proposed DNN framework in Ref. [46] and comparison of three convolution operations.

the training set via multiview triangulation. A High-Resolution Net (HRNet) [52] is then trained to regress the location of projected 2D corner point landmarks on the spacecraft from the input greyscale image. Finally, the optimal poses are obtained by the proposed geometric optimisation algorithm based on simulated annealing, where the initial pose is estimated from a PnP solver.

More specifically, the proposed DNN framework contains two modules. The first uses an HRNet front-end/Faster R-CNN combination to detect the 2D bounding box of the target in the input image. The RoI is then cropped and resized for use in the second model, which consists of a pure HRNet and is trained on an MSE loss between the predicted and ground truth heatmaps of the visible landmarks in each image.

To achieve a fast and accurate estimate of the pose, Huo et al. [53] developed a novel DLs-based approach combining PnP and geometric optimisation. A new and lightweight tiny-YOLOv3 based framework is designed to predict the 2D locations of the projected keypoints of the constructed 3D model. Fig. 10 shows the corresponding regression network, in which the output of tiny-YOLOv3 is modified to establish a box reliability judgement mode for detecting the S/C and predicting the 2D RoI. Next, the regression of S/C keypoints is achieved by replacing the FCLs

with CLs to yield heatmaps. Finally,  $PnP$  and bundle adjustment are utilised to generate the initial pose and optimise it, respectively, which improves the accuracy and robustness of the proposed approach. Their method is evaluated on the SPEED dataset and achieves competitive performance in spacecraft pose estimation with a lighter computational footprint.

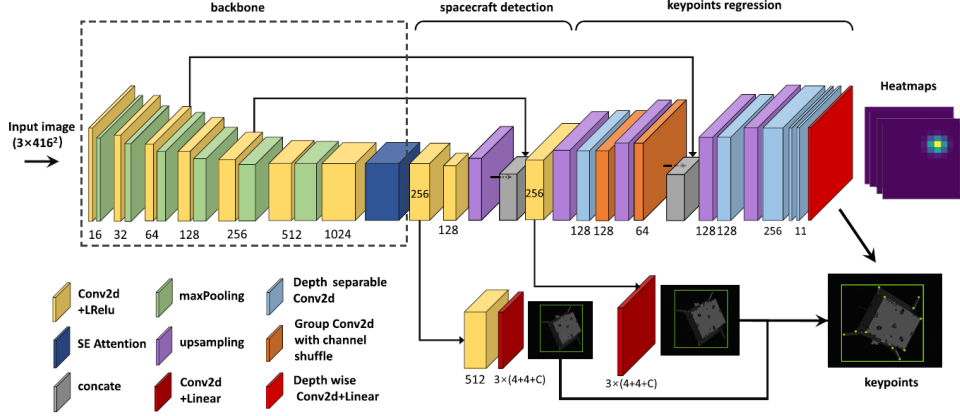


Figure 10: The overall structure of the network designed by Huo et al. [53].

Another indirect DLs-based scheme combines a CNN-based feature detector with a  $PnP$  solver and an Extended Kalman Filter (EKF) to guarantee a robust pose estimation [27]. The authors build an hourglass-shaped CNN composed of a six-block encoder and a six-block decoder to estimate the heatmaps of 16 predefined corners on the Envisat spacecraft. A target detection module is not incorporated since the presence of Earth in the background is not considered. Using the weights from the heatmaps, an associated landmark covariance is calculated. Two testing campaigns are then performed. The first one uses a dataset composed of singular images and computes the relative pose by incorporating the covariance of the regressed landmarks into the  $PnP$  procedure [54]. The second campaign considers a sequential dataset simulating a V-bar approach with Envisat at a fixed relative distance, where the target performs a roll rotation with respect to the Local-Vertical, Local-Horizontal (LVLH) frame of reference. The relative pose is estimated by a tightly coupled EKF based on a Clohessy-Wiltshire dynamical model. Sensor measurements input of the filter, the landmark locations and covariances, come from the CNN. The filter achieves steady-state position errors inferior to 0.2 m for all axes, and the attitude errors are under 2 deg.

A pipeline similar to Ref. [53] is investigated by Huan et al. [21], achieving results nearly an order of magnitude better in the precision and accuracy of position and attitude estimation relative to the

SPN framework. The training methodology consists of four steps: 1) manual selection of images from the training dataset to be used for the reconstruction of the target 3D model, 2) detection of the 2D bounding box by an ODN, 3) estimation of the 2D image location of keypoints from a KRN, and 4) projection of the 3D groundtruth keypoints onto the image plane and solving the PnP problem from the correspondences with the estimated keypoints. Differing from Ref. [53], the proposed target detection network and KRN in Ref. [21] apply the state-of-the-art HRNet as backbone. The 6-DoF pose is finally predicted by non-linear minimisation of a Huber reprojection loss. The training dataset is constructed of synthesised greyscale images, and the test set images are captured in real-time using a monocular camera.

Shi et al. [6] transfer the state-of-the-art CNN techniques to target CubeSat detection, but with no further discussion on pose estimation. Inception-ResNet-V2 [55] and ResNet-101 [37] are combined and trained to estimate the bounding box of the target S/C with a laboratory test platform. The final FCL of ResNet-101 is reduced to two classes to differentiate between the "1U-CubeSat" and "3U-CubeSat" labels. The pre-trained weights of the CNN are obtained from the COCO dataset [56] and further trained on a mixture of real and synthetic CubeSat images. Their simulation results indicate that the Inception-ResNet-V2 framework achieves a slightly higher accuracy and precision for S/C detection, whereas the ResNet-101 network is less computationally heavy. To tackle a similar problem, Ming [57] constructs a feature extraction network and Region Proposal Network (RPN) structure framework based on Faster R-CNN [32] on the CNN Caffe [58] open platform. The proposed network performs intelligent identification of a spacecraft module in the image sequence and filters out the certain components of interest.

For other space missions beyond rendezvous, Yi [59] studies the relative position estimation problem of a docking mission (below 10 m). Assuming relative attitude has been adjusted, the method utilises a modified VGG-16 to regress the relative position between docking rings. Further position smoothing and relative speed estimation are achieved by Kalman filtering. Additionally, a satellite positioning error compensation technique based on DL is discussed by Jiaming [60]. Large amounts of data are collected to generate a robust model; a CNN, a depth belief network, and a Recurrent Neural Network (RNN) are trained on satellite location data are collected by the Institute of Technology of the Chinese Academy of Sciences, which aims to generate a robust model. A CNN, a depth belief network, and a RNN are trained on the collected data, of which the

Table 2: Summary of DL-based indirect pose estimation methods for spacecraft relative navigation.

Ref.	Backbone	Transfer learning	Type	Dataset	Pose estimation by
[46]	YOLO, MobileNet	✗	Keypoint regressor	Synthetic (NST <sup>1</sup> ), SPEED	PnP
[20]	HRNet + Faster R-CNN	✗	Keypoint regressor	SPEED	PnP
[53]	tiny-YOLOv3	COCO	Keypoint regressor	SPEED	PnP
[27]	Hourglass network	✗	Keypoint regressor	Synthetic (Cinema 4D, Envisat S/C)	PnP + EKF
[21]	2-layer CNN	✗	Keypoint regressor	Synthetic, real (lab)	PnP
[59]	VGG-16	✗	Position regressor	Synthetic (Blender)	EKF
[6]	ResNet	COCO	Classifier	Synthetic, real	CubeSat detection
[57]	Faster R-CNN	✗	Regressor	Synthetic	Object detection
[60]	LSTM	✗	Regressor	Real	Position error compensation

<sup>1</sup> Neural Style Transfer; applied to randomise the texture of the spacecraft.

CNN performs the best compensation result.

Table 2 contains a brief summary of indirect DL-based algorithms for spacecraft pose estimation and related applications. As illustrated, the earlier studies (last four referenced) are more focused on parts of pose estimation missions, such as detection and position estimation. PnP and EKF are commonly combined to refine poses output by DNNs. Moreover, transfer learning and very deep networks are rarely utilised when DL methods are combined with optimisers. This could potentially be due to the fact that the pipelines rely heavily on these optimisation steps at the end, which are able to guarantee a decent estimate of the pose. In this way, shallow networks can reduce the computational cost, which is beneficial for practical use and potential onboard implementations.

### 3. Crater and Hazard Detection for Terrain Navigation Using DL

Exploring and landing on the lunar surface has long been a challenge of great interest within space technology and science. Recent developments in DL have led to a renewed interest in learning-based TRN. Craters are ideal landmarks for relative navigation on or around the Moon and asteroids [61, 62]. Additionally, hazards should be avoided for a successful landing mission. This section, therefore, reviews the field of DL-based terrain navigation in three aspects: crater detection, hazard detection, and TRN methods, all using DNNs. Figure 11 shows schematic diagram and difference between the three aspects.

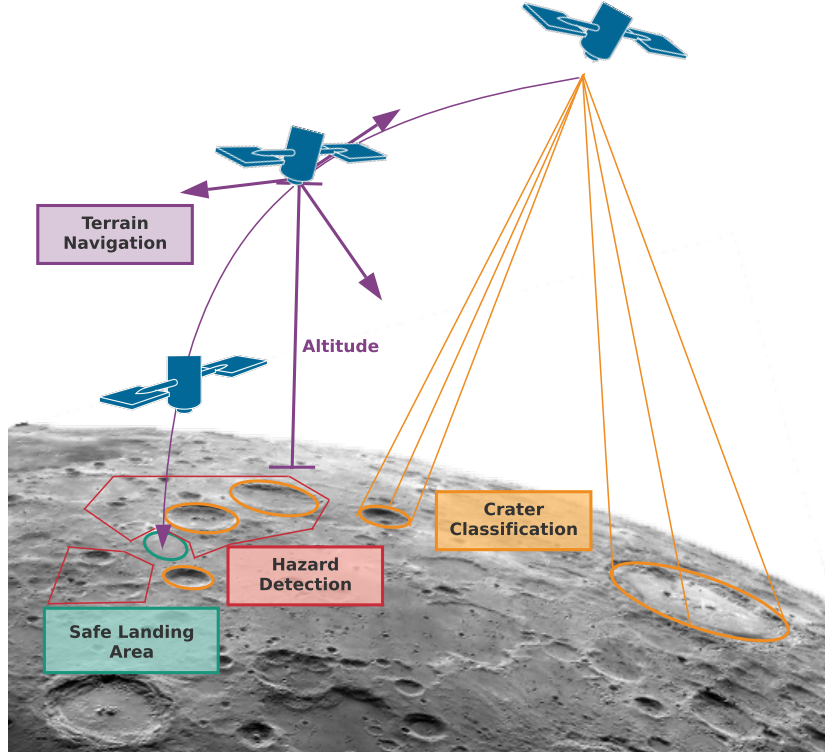


Figure 11: Scenarios of crater detection, hazard detection for safe landing area, and terrain navigation.

### 3.1. Crater detection

With advances in computer vision and successful applications of CNNs in the object detection area, CNN-based crater detection algorithms are also emerging. However, most of the earlier methods only utilise a CNN as a classifier to validate selected features, such as in Refs. [63, 64, 65]. The shapes of natural craters vary in morphology, including peak rings, central pits, and wall terraces [66]. Some craters may also overlap with others. Considering the illumination conditions and different poses of on-board cameras, the imaged craters can be diverse in terms of dimensions and appearance [61]. Conversely, robust crater detection algorithms have been developed by applying DNNs to fully process raw crater images, exhibiting promising results which have attracted a lot of interest.

The Python Crater Detection Algorithm (PyCDA) [67] is an open-source crater detection library composed of a detector, extractor, and classifier, which focuses on detecting new craters that have never been catalogued. PyCDA uses a downsized U-Net architecture to compute the

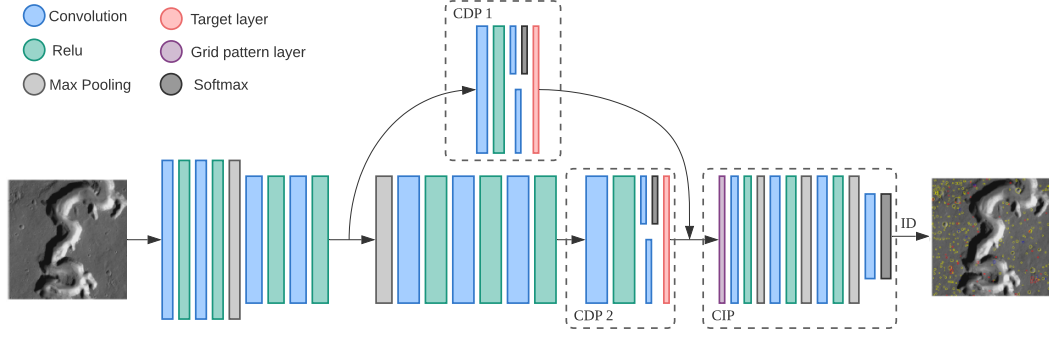


Figure 12: Framework of CraterIDNet reproduced from Wang et al. [61].

per-pixel likelihoods of a crater rim from inputs of greyscale intensity images. The pixel prediction map is then fed to the extractor to generate a list of crater candidates. A classifier CNN is finally applied to determine true craters. Thanks to PyCDA, a considerable amount of craters have been detected and categorised, thus helping to generate new labelled datasets for training and testing of DL algorithms.

Wang et al. [61] proposed an end-to-end fully CNN, CraterIDNet, for simultaneous crater detection and identification. CraterIDNet takes remote sensing images in various sizes and outputs detected crater positions, apparent diameters, and indices of the identified craters. Instead of using large off-the-shelf DNN models, a small CNN architecture pre-trained on Martian crater samples [68] is first developed to extract feature maps. Next, two pipelines, namely Crater Detection (CD) and Crater Identification (CI) are proposed for simultaneous detecting and identifying craters. The CD process involves detecting the presence of craters and locating them within the image if they exist. The output of CD is then fed to the CI process to match the detected craters to surface landmarks in a known database, and matches of CI will provide position estimation. Fig. 12 shows the whole framework of CraterIDNet. The CD modifies the RPN architecture [32] as the backbone, regressing objectness scores and crater diameters from feature maps. Due to different craters sizes, two CD pipelines are designed by sharing same CLs but with different parameters. Later, craters are identified by CI that combines a proposed grid pattern layer and CNN framework. For the training and testing dataset, 1600 craters are manually catalogued and enlarged to a final sample set of 16 000 instances through data augmentation. Experiments reveal that the light CraterIDNet with a size of 4 MB performs better than previous algorithms [64].



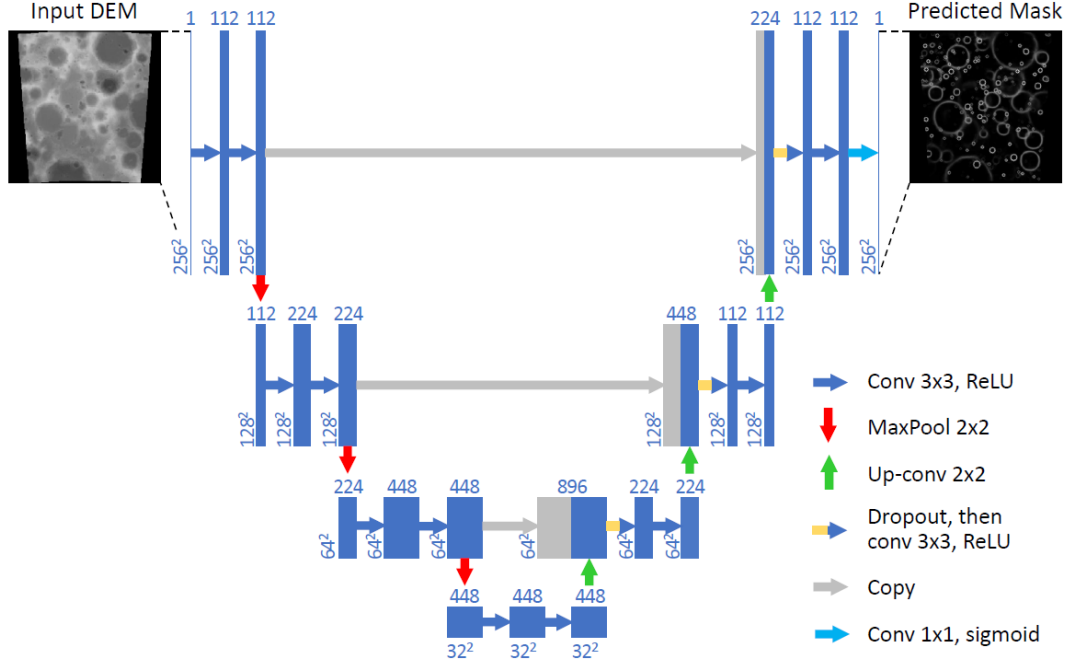


Figure 13: Architecture of DeepMoon network [69]

Silburt et al. [69] employ a CNN architecture for robust crater detection on the lunar surface using DEMs. The method relies on the developed DeepMoon network to identify the craters in terms of their centroid and radius, and outputs pixel-wise confidence maps of crater rims on the surface of a rocky body. DeepMoon modifies U-Net [70] by changing the input image size, the number of filters in each convolution layer, and the use of dropout [71] in the expansive path for memory limitations and regularisation respectively. Fig. 13 presents the DeepMoon architecture.

For training, the data used in DeepMoon is generated by merging two human-generated crater catalogues, which are the Lunar Reconnaissance Orbiter (LRO) Wide Angle Camera (WAC) Global Lunar DEM [72] and the LRO Lunar Orbiter Laser Altimeter DEM [73]. The dataset is split into equal train-validation-test parts, yielding 30 000 DEM images per part. The minimised loss function is chosen as the pixel-wise Binary Cross-Entropy (BCE). DeepMoon produces a crater rim prediction mask, which is then fed to a low-level image process and a template matching procedure to determine the actual craters. The median fractional longitude, latitude and radius errors are 11 % or less, representing good agreement with the human-generated datasets. Additionally, transfer learning from training on lunar maps to testing on maps of Mercury is qualitatively demonstrated

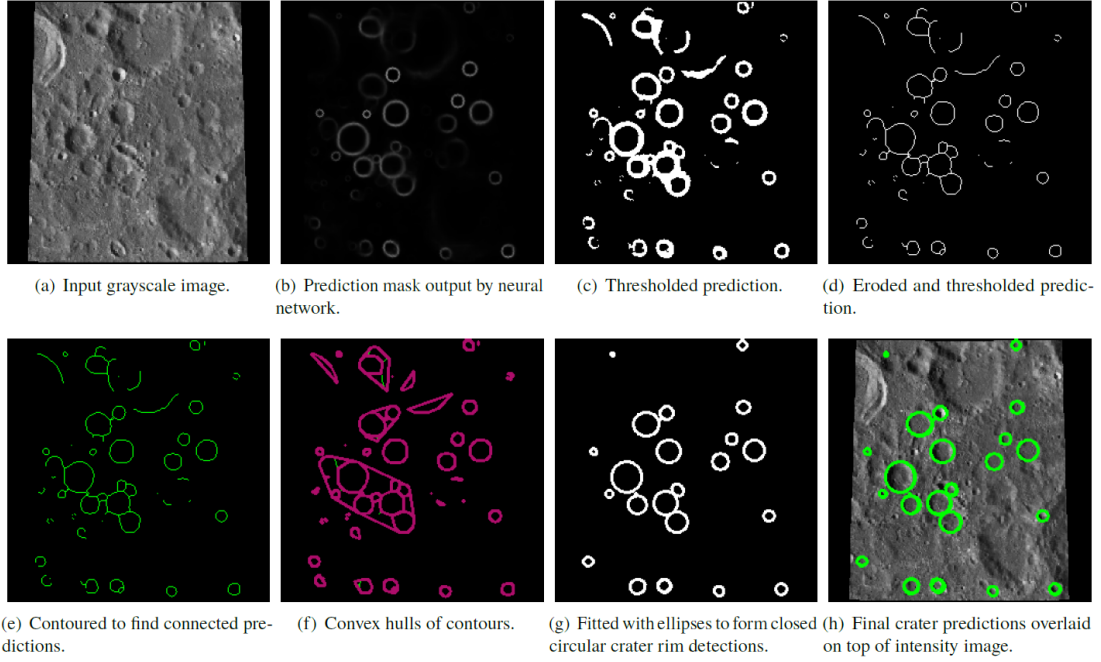


Figure 14: Feature extraction steps of LunaNet [25]

successfully.

Downes et al. [25] propose the LunaNet framework to detect craters for lunar TRN, which is quite similar to DeepMoon with the exception that it takes greyscale images as inputs. Thus, the method is more suitable for implementation aboard a spacecraft equipped with an optical camera without the need for a depth sensor. The output of the CNN is, like DeepMoon, a crater rim prediction mask. However, the craters are extracted through a different method and, Fig. 14 shows each feature extraction step of the LunaNet, including prediction mask, eroded and thresholded prediction, contour detection, and ellipse fitting. The data preparation is also akin to the process followed by DeepMoon, with the LRO WAC Global Lunar DEM dataset [72], followed by a histogram rescaling of the input greyscale images to match the intensity distribution of a DEM image. Based on the pre-trained DeepMoon weights, LunaNet reduces the training effort and final detection results. Experimental results indicate that LunaNet’s performance surpasses DeepMoon and PyCDA in terms of robustness to noisy images, location accuracy, and average crater detection time.

It has been observed that areas with low solar angles, where there is heavy shadowing, result in reduced crater detection reliability. Lee et al. [74] employ a CNN-based object detector to

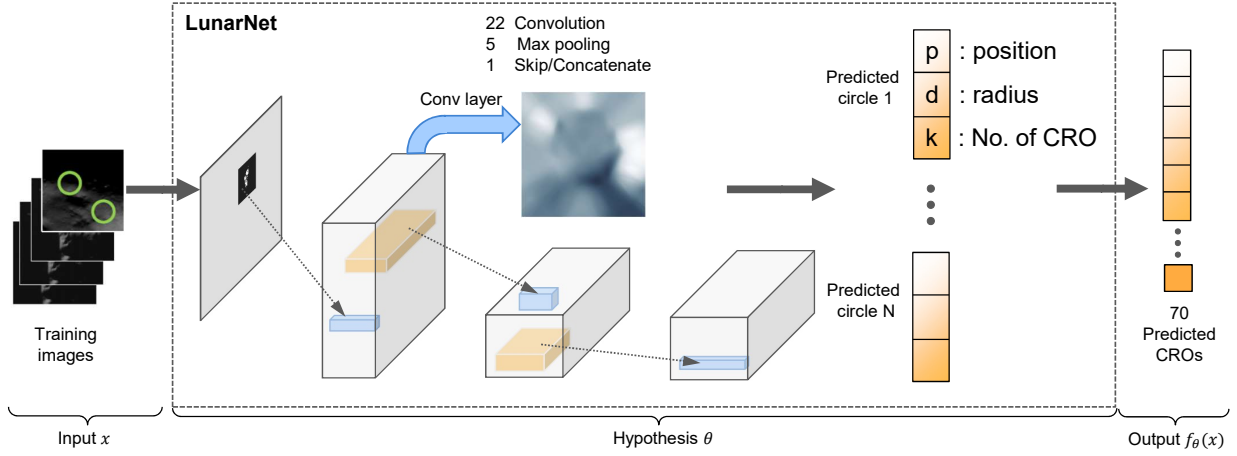


Figure 15: CNN-based CRO discriminator (LunarNet) [74]

distinguish likely landmark candidates and predict detection probabilities along various lighting geometric flight paths, aiming to identify high-value landmarks by using optical navigation systems.

435 A massive dataset based on real lunar-surface data is collected. A Candidate for a Regional Object (CRO) is defined as an image object with specific latitudes and longitudes. The LunarNet architecture (Fig. 15; see also the process of LunarNet-based landmark selection in Ref. [74]) is then used and trained to identify CROs by maximising the discrimination between local areas of the Moon. Finally, the CRO performance map is formed based on the scored CROs arranged by  
 440 considering the azimuth and elevation angles of the Sun during the year. Numerical experimental results demonstrate that the proposed landmark detection pipeline can provide usable navigation information even at Sun angle elevations of less than 1.8deg in highland areas, which indicates a successful application for the worst dark highlands near the South Pole.

### 3.2. Hazard detection

445 Hazard detection is considered the vital research field of space TRN to avoid failures during landing. In 1974, Apollo program officials introduced manual hazard and target selection for lunar descent guidance [75]. In the past decade, many algorithms have been developed which benefit from the increasing computational power of processor devices. Since 2006, promoted by the Autonomous Landing Hazard Avoidance Technology (ALHAT) project [76] conducted by the National Aero-  
 450 nautics and Space Administration (NASA), there has been growing interest in hazard estimation based on DEMs. In 2012, Furfaro et al. [77] implemented an AI system to autonomously select a

soft landing site in a region with safe terrains for Venus and Titan. In 2015, Maturana and Scherer [78] used what they called a 3D-CNN to create a safety map for autonomous landing zone detection for helicopters.

Earlier research in NN-based Hazard Detection and Avoidance (HDA) for lunar landing is studied by Lunghi and Lavagna [79] and Lunghi et al. [80], who demonstrate the ability and attractive properties of Artificial Neural Networks (ANN) for real-time applications. The ground truth is calculated from the corresponding DEM by thresholding pixel-wise figures. Input images of the terrain are manually processed at a resolution of  $1024 \text{ px} \times 1024 \text{ px}$  to extract a 13-dimensional vector per pixel comprising the image intensity mean, standard deviation, gradient and the Laplacian of Gaussian (LoG) at three different scales, and the Sun’s inclination angle. Following this, the crafted features are fed to a neural network, outputting a  $256 \text{ px} \times 256 \text{ px}$  hazard map with each pixel value denoting a confidence value. From the output hazard map, candidate landing sites are obtained via pixel thresholding and scored global landing potential by analysing minimum radial dimension requirements, distance to an a priori nominal landing site, and the NN scores of pixels inside the candidate radius. The target landing site is selected as the one that maximises the global score. Two different pipelines are developed: one based on a Multilayer Perceptron (MLP) with 15 nodes, and the other based on a cascading NN with successive layers of hidden information added during training. A test set of 8 images including four landscapes in two Sun inclination angles are utilised to evaluate two proposed pipelines. The predicted hazard maps during training have a negligible difference, with 0.0194 for the MLP and 0.02039 for the cascade of pixel-wise MSE. However, the former proved better at determining safe landing sites. In addition, qualitative results have been presented for asteroid images acquired by the Rosetta probe.

Recently, Moghe and Zanetti [81] presented a more modern approach towards tackling the same problem. Aiming at the hazard detection of the ALHAT project, the authors implement an hourglass-like CNN architecture with copy and crop connections based on U-Net [70]. The framework processes DEMs directly and classifies safe and hazardous landing spots with the output map. Through data augmentation and transforming existing datasets, they create a new dataset from the LRO dataset [63]. The output, similarly to Ref. [80], is a confidence map followed by a threshold to yield a binary landing/non-landing score, despite not provide a specific landing site. Results on a set of 100 testing images demonstrate an average hazard mapping Dice accuracy score

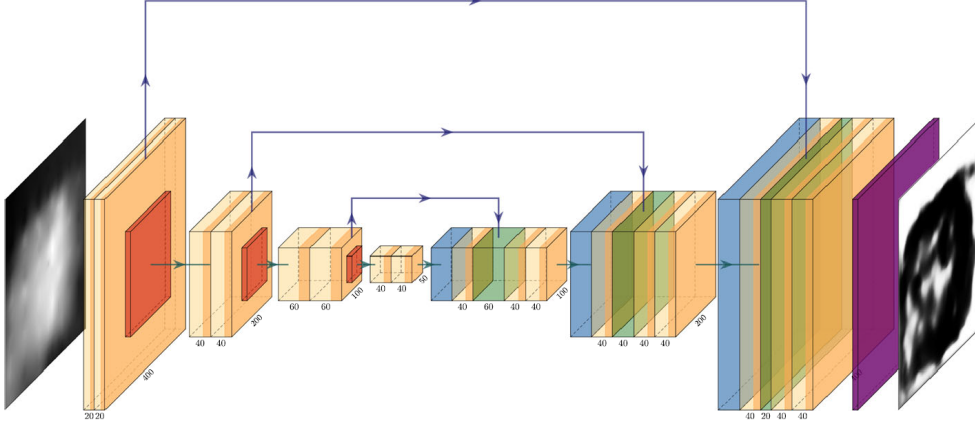


Figure 16: The topology of the network in [82]

of 83 % and indicate the potential of real-time processing in future missions. Later, Moghe and Zanetti [81] expand and modify their work in Ref. [82], using the same network architecture but featuring improved layers covering the input size, output size, and layer width. The topology of the modified network is illustrated in Fig. 16. Similarly, the Albumentations data augmentation suite [83] is used to prepare data. The modified CNN-based network outputs a mean pixel accuracy of  $\sim 92$  % on the same testing dataset of Ref. [81].

### 3.3. Terrain navigation

In the field of image-based planetary TRN, Campbell et al. [28] first utilise a CNN architecture trained on a series of images rendered from a DEM simulating the Apollo 16 landing site to output the position of a spacecraft relative to the ground along one direction. The problem is posed by taking a centre 128 px wide strip from the original  $1024 \text{ px} \times 1024 \text{ px}$  nadir base image, considering each pixel location along the on-track dimension as its own class.  $128 \text{ px} \times 128 \text{ px}$  training images are generated by sampling every 8 px horizontally across the strip and rendering it 11 times at different Sun illumination angles. The 1024-dimensional one-hot vector, which labels the position along the track line, is then applied to each image. The CNN is composed of three CLs and each followed by a max pooling layer. Thirty images are rendered at unseen Sun angles to make up a test dataset. Six of these are classified correctly, while in general, the maximum error observed is equal to 5 px. For a ground sample distance of 0.5 m, this means that achieved position errors are bounded at 2.5 m. The testing is repeated for training images resampled at 4 px, and the errors

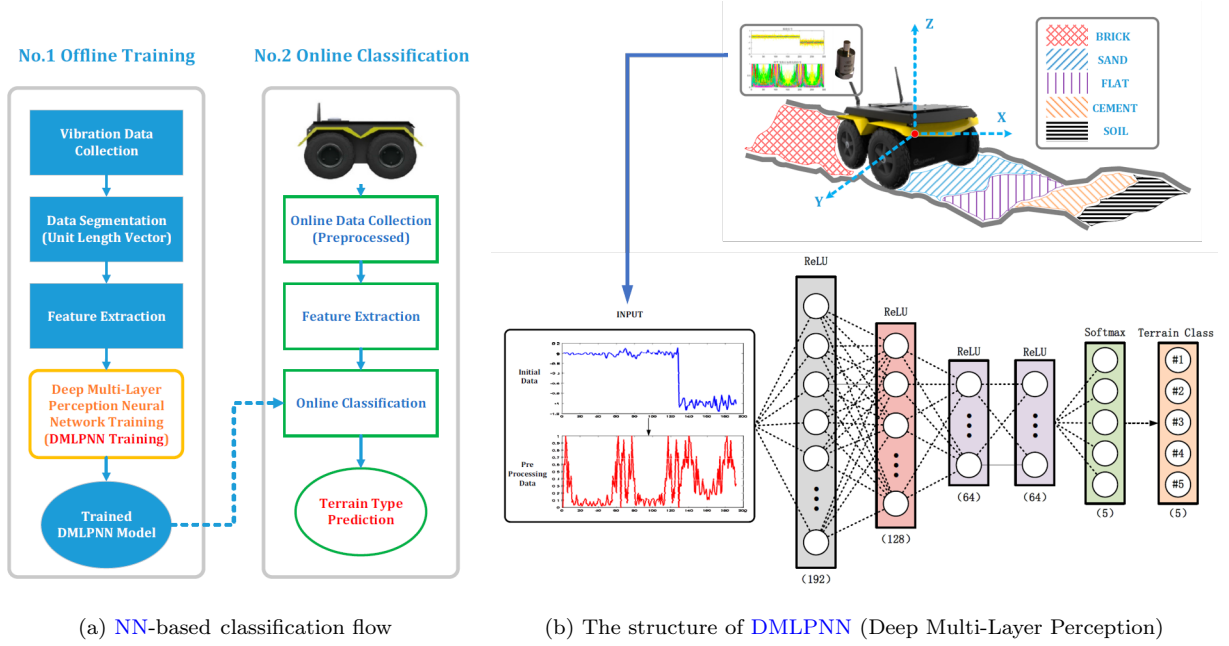


Figure 17: DMLPNN classification flow and proposed architecture of Bai et al. [85].

dropped to a maximum of 3 px (or 1.5 m).

In 2020, Downes et al. [26] explored how their LunaNet could be applied to the TRN problem and reported a system for the robust estimation of relative position and velocity information. Thus, LunaNet is utilised to detect and match craters to known lunar landmarks from frame to frame across a trajectory. The matched craters are treated as features feeding to a feature-based EKF, where the state of the filter is the position and velocity of the camera in Lunar-Centred, Lunar-Fixed Coordinates (LCLF), as well as the location of detected features in this same reference frame. Compared to an image processing-based crater detection method [84], the LunaNet + EKF combination produces considerable improvements on the accuracy of the TRN, with reliable performance in variable lighting conditions.

Accurately identifying the detected terrain environment helps to achieve successful missions relying on planetary rovers. However, vision-based TRN systems are difficult to effectively perceive the material and mechanical characteristics of the terrain environment. Thus, Bai et al. [85] and Chengchao [86] investigate several terrain classification and recognition methods from vibration using DNNs. The experimental and NN classification flows are illustrated in Fig. 17a. The authors compare three different learning-based approaches towards terrain material perception and classi-

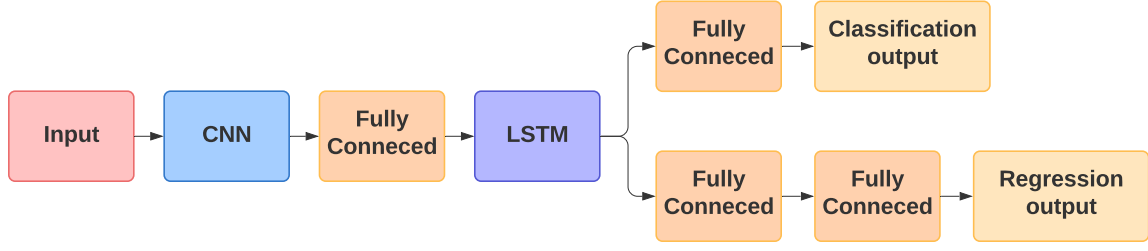


Figure 18: The architecture proposed by Furfaro et al. [87]

fication: an improved NN algorithm, a DMLPNN algorithm, and CNN-LSTM based algorithm. Among these three schemes, the DMLPNN achieves the best performance [85]. To classify textures, DMLPNN (shown in Fig. 17b) adopts a five-FCL architecture, in which the activation functions are ReLU and softmax for the first four and last layers, respectively. For the dataset and training, three-dimensional raw vibration data collected by sensors is first segmented to a vector with a fixed duration. Using the fast Fourier transform, the vector is then transferred to the frequency domain, in which the eigenvectors are obtained for network training. Five different textures, including brick, sand, flat, cement, soil, are trained and recognised by DMLPNN with high overall classification accuracy.

For an autonomous lunar landing scenario, Furfaro et al. [87] propose a DNN architecture that predicts the fuel-optimal control actions only using raw greyscale images taken by an on-board lander camera (Fig. 18). The architecture is a five-layer CNN with three sequential images as input for each timestep. The DNN is modified with an LSTM back-end connected to two further branches: one for regression and one for classification. For training the network, a set of optimal trajectories is computed numerically via Gauss pseudo-spectral sampling methods using the General Purpose Optimal Control Software (GPOPS II) [88], producing a set of initial and final relative positions and velocities. Each state of the optimum trajectory is simulated by raytracing a DEM of a patch on the Lunar surface, resulting in 562 images with  $256 \text{ px} \times 256 \text{ px}$  of resolution. For better performance, the model is retrained explicitly with subsets of data that do not produce satisfactory results on the first try [89].

### 3.4. Brief summary

The use of DNNs in crater and hazard detection has not been widely investigated due to the lack of labelled databases. Datasets containing crater images which are open to the public do exist, e.g. the

540 Lunar Crater Database<sup>6</sup> [63] or the Robbins Mars Crater Database [68]. Yet, manually catalogued craters are required for applying supervised DL methods as presented in [61, 90, 25, 72, 73]. There exists still a gap towards the automatic generation of DL crater datasets, and in the past two years there have been increasing studies of DNNs with promising performance for TRN tasks.

## 4. DL-based Relative Navigation for Asteroid Research

### 545 4.1. Challenges and Motivations for DL-based Asteroid Exploration

Recent trends in small planetary exploration have led to a proliferation of studies that include asteroids and comets, pushed by scientific, planetary defence, and resource exploitation motivations [91, 92]. Autonomous navigation is demanded due to the long communication delay and complicated dynamic environment in the vicinity of asteroids [93]. Thus, it becomes necessary to develop  
550 new autonomous navigation algorithms for future asteroid sample and return missions, for which DL techniques may provide a potential alternative.

The aforementioned studies demonstrate the potential of DNNs for image patch classification invariant under illumination changes applied to terrain navigation. The same principle could be used for other relative navigation applications, such as asteroid location pinpointing, illustrated in  
555 Fig. 2b. Near-Earth Asteroid (NEA) missions, however, are more challenging than lunar missions; this is because one has limited information on the gravitation and environment of asteroids. If the celestial body and its orbit environment are in great uncertainty, all plans elaborated on-ground may dramatically fail when implemented in space [94]. Additionally, the lack of labelled ground truth data for asteroids challenges the application and development of DL techniques in asteroid  
560 detection and landing [95].

### 4.2. Previous Works Contributing to the Field

For asteroid missions, earlier researchers have made various contributions towards NN-based orbit and dynamics uncertainty estimation. Harl et al. [96] develop a NN-based state observer to estimate gravitational uncertainties that spacecraft experience in an asteroid orbiting scenario. The NN of  
565 the proposed state observer outputs the uncertainty as a function of the states instead of discrete

---

<sup>6</sup>[https://astrogeology.usgs.gov/search/map/Moon/Research/Craters/lunar\\_crater\\_database\\_robbins\\_2018](https://astrogeology.usgs.gov/search/map/Moon/Research/Craters/lunar_crater_database_robbins_2018).



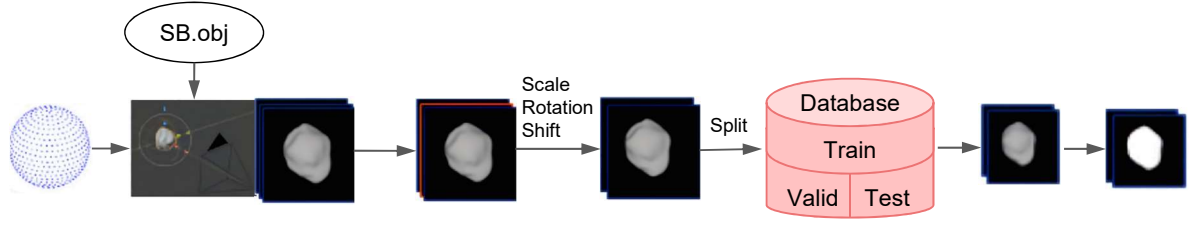
values of an [EKF](#). Guffanti [94] trains a neural network as an autonomous motion planning unit to compute the optimal spacecraft orbital configuration, which takes the uncertain [NEA](#) dynamics parameters created by navigation filters and the selected trade-off. Song et al. [97] also employ a six-hidden-layer [DNN](#) to quickly estimate the gravity and gradient of irregular asteroids and further apply the [DNN](#)-based gravitational model in orbital dynamic analysis. Instead of focusing on-orbit estimation, Kalita et al. [98] introduce an [NN](#) to the formulation of asteroid missions in terms of the planning and design phases, while Feruglio et al. [99] utilise a feed-forward [NN](#) to autonomously identify a [S/C](#) impact event. Viavattene and Ceriotti [100] take advantage of a [NN](#) to map the transfer time and cost for [NEA](#) rendezvous trajectory.

[DNN](#)-based optical navigation is an increasingly important area in asteroid exploration missions, which can manage challenges of previous schemes, including traditional high-cost and high-risk spacecraft systems, irregular and illuminated asteroids, and conventional image processing techniques. In such a scenario (Fig. 2b), the chaser may be commanded to inspect a particular patch on the surface of the asteroid it has rendezvoused with (observed on frame  $\mathfrak{F}_c$ ), which is intrinsically a localisation task requiring the estimation of  $\mathbf{T}_{ct}$ . If the asteroid has been previously mapped, and there exists a codebook with annotated landmarks (on frame  $\mathfrak{F}_t$ ) for comparison, there are two possible approaches. The first follows the same direct classification procedure as Ref. [28], where a [DNN](#) is used to match the observed patch with the corresponding patch in the codebook, which is annotated with the relative pose, but with the dataset of Lunar surface. The alternative approach is to have a single class per patch on the database and train the [DNN](#) to be robust to viewpoint distortion, and then rely on classical image processing techniques to infer the pose based on the different observed features between the observations and matched patches.

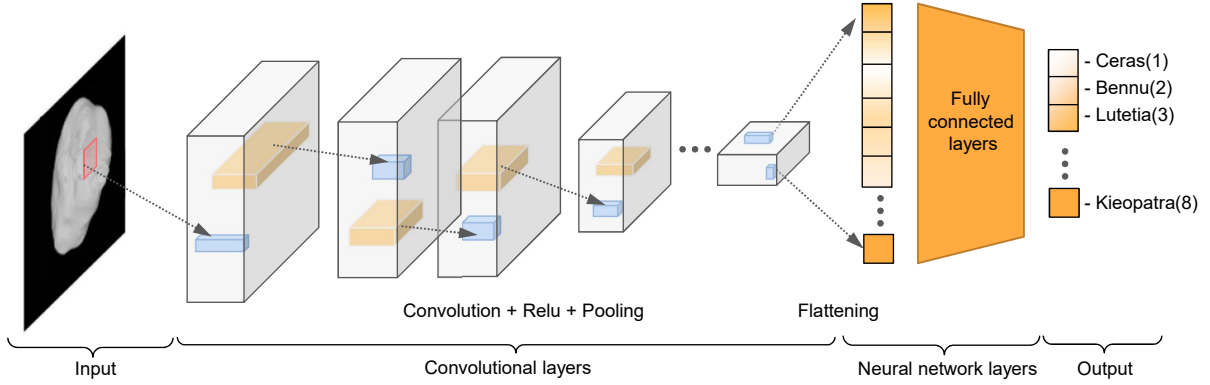
Pugliatti and Topputo [101] first present [CNN](#)-based methods for on-board small-body shape classification since shape information can enhance the image processing and autonomy of self-task planning. A set of 8 well-known models from the Planetary Data System ([PDS](#)) node<sup>7</sup> is selected to represent the most important features of small asteroids at a global scale. Fig. 19 presents a sketch of the steps for building the database and the proposed [CNN](#) framework for classifying asteroids. The database is generated in Blender with an assumed camera pointing and illumination, and further augmented in TensorFlow [102] with random rotations, translations, and scaling. The

---

<sup>7</sup><https://sbn.psi.edu/pds/shape-models>.



(a) Step flows for database generation



(b) Schematic representation of the proposed CNN

Figure 19: Database generation flow and proposed CNN architecture of Pugliatti and Topputo [101].

database composed of 20 988 images is divided into training, validation and test sets according to a 80 %-10 %-10 % split. Their CNN architecture has five CLs in sequence, with each followed by a pooling layer, a reshape operation, and three FCLs to classify. A hyperparameter search is used to obtain network parameters. Three traditional approaches, such as Hu invariant moments, Fourier descriptors, and polar outlines, are compared, in which the proposed CNN-based scheme performs best.

Later, Pugliatti and Topputo [91] proposed on-board autonomous navigation using segmentation maps and a CNN to estimate spacecraft position concerning an asteroid fixed reference frame. The CNN transferred from the MobileNetV2 network [48] classifies the segmentation maps to generate a rough estimate of the position information from the input. The relative position is finally obtained by refining the output of the CNN using an advanced normalised cross-correlation method. Didymos and Hartley are selected as representatives of regular and irregular small-bodies to create the dataset, which includes 49 716 samples of synthetic maps for five different scenarios. Experimental results indicate the capability of CNN in predicting the correct class and achieve a

relative position error below 5 %–8 % of the range from the asteroid.

In 2021, Ravani et al. [95] developed a novel Mask-Region CNN to detect landing sites for autonomous soft-landing on asteroids. Since there is no open public dataset of potential landing sites labelled with ground truth, the authors first gather image mosaics of the asteroid Vesta from the PDS of NASA<sup>8</sup> and then fragment the large images into smaller ones with a fixed size. Next, the training dataset is labelled manually. For the Mask-Region CNN pipeline, it follows a backbone network of ResNet-50-C4 for initialisation; an RPN for extracting feature maps; Faster R-CNN for RoI alignment; the network head is structured using FCLs for computing the bounding box; and the mask head of a FCL network [103] is used for calculating the pixel-level mask. Comparing with conventional image processing methods, the proposed network on their dataset results in an accuracy of 94 % with lower computational time cost in the implementation phase.

## 5. Summary and Conclusion

This work surveyed recent trends in deep learning techniques for 6-DoF relative pose estimation in spaceborne applications. Contributions in the field of computer vision were presented, followed by concrete applications from the literature to autonomous spacecraft navigation, including spaceborne pose estimation, crater and hazard detection of terrain relative navigation, and DL-based asteroid navigation. This survey is motivated by the applicability of DL techniques in relative spacecraft navigation for future space missions, i.e. rendezvous, docking, formation flying, descent and landing on the lunar surface, orbiting and inspecting asteroids. The general DNN framework for the applications in this research area was reviewed in terms of network structure, type of network, training method, dataset topology and generation, and attained performance.

First, a review of DNN-based S/C relative pose estimation techniques was given, in which a top level distinction between supervised and unsupervised methods was made, whereby contributions in the space domain were found to belong exclusively to the former. Context in terms of preceding ground-based applications was established. Further lower level categorisations were made; in particular, it was found that many techniques favoured a direct approach (so called “end-to-end”), where a DNN pipeline is trained directly on images to yield the relative state. Indeed, this is a very appealing property of deep learning, as not only is the feature extraction task relayed to a

---

<sup>8</sup><https://pds.nasa.gov>.

CNN, but so is the modelling task, eliminating the "middleman" and allowing the user to focus mainly on the architecture design and optimisation of learnable parameters. However, it was seen that more accurate solutions were obtained by combining them with classical methods. For these indirect methods, a CNN was tasked with regressing the locations of 2D keypoints on the target and estimating the relative pose from geometrical correspondences with their 3D counterparts, using techniques such as PnP or nonlinear optimisation. Furthermore, such solutions are easily incorporated into navigation filters to further refine the estimate with continuous, smooth consistency (also beyond pose estimation). The role of RNNs, particularly LSTMs, is highlighted in the processing of a continuous stream of images. Tables 1 and 2 summarises these findings in terms of relative pose estimation error for spacecraft rendezvous DL applications.

Second, the applications of DNNs to TRN were divided into three aspects for surveying, in which the DNN-based crater and hazard detection methods were recognised as contributors towards building a terrain navigation system. It was pointed out that public open data for training and testing of DNN-based TRN frameworks is limited. Furthermore, DL-based relative navigation methods focusing on asteroid missions were provided. The challenges and motivations were discussed before a detailed review of this field.

Lastly, regarding unsupervised learning methods (i.e. concerning cases in which the desired output for each input is not given during training), far too little attention has been paid to this kind of technique for space navigation. However, unsupervised techniques such as CNN-SLAM (Simultaneous Localisation and Mapping) or unsupervised VO are underlined as a potential novel approach for the space domain and may be investigated in future. Additionally, most publications study the application of DL in space in a theoretical way without being concerned with computational performance; indeed, only a few publications [19, 22, 81] focus on actual deployments on hardware, considering things like execution time, and size of the training dataset. Therefore, it can be concluded that these studies towards the actual engineering practice have been little discussed and require further development.

## References

- [1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.
- [2] Bong Wie, Vaios Lappas, and Jesús Gil-Fernández. Attitude and orbit control systems. In Malcolm Macdonald

and Viorel Badescu, editors, *The International Handbook of Space Technology*, chapter 12, page 362. Springer Praxis Books, 2014. doi:[10.1007/978-3-642-41101-4](https://doi.org/10.1007/978-3-642-41101-4).

- [3] Shyam Bhaskaran, S Desai, P Dumont, B Kennedy, G Null, W Owen Jr, J Riedel, S Synnott, and R Werner. Orbit determination performance evaluation of the deep space 1 autonomous navigation system. In *AAS/AIAA Space Flight Mechanics Meeting*, 1998. URL <https://trs.jpl.nasa.gov/bitstream/handle/2014/19040/98-0222.pdf?sequence=1>.
- [4] Vivek Kothari, Edgar Liberis, and Nicholas D. Lane. The final frontier: Deep learning in space. In *Proceedings of the 21st International Workshop on Mobile Computing Systems and Applications*, HotMobile '20, page 45–49, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450371162. doi:[10.1145/3376897.3377864](https://doi.org/10.1145/3376897.3377864).
- [5] Lorenzo Pasqualetto Cassinis, Robert Fonod, and Eberhard Gill. Review of the robustness and applicability of monocular pose estimation systems for relative navigation with an uncooperative spacecraft. *Progress in Aerospace Sciences*, 110:100548, 2019. ISSN 0376-0421. doi:<https://doi.org/10.1016/j.paerosci.2019.05.008>.
- [6] Jian Feng Shi, Steve Ulrich, and Stéphane Ruel. Cubesat simulation and detection using monocular camera images and convolutional neural networks. In *2018 AIAA Guidance, Navigation, and Control Conference*, page 1604, 2018. doi:[10.2514/6.2018-1604](https://doi.org/10.2514/6.2018-1604).
- [7] Sumant Sharma, Connor Beierle, and Simone D'Amico. Pose estimation for non-cooperative spacecraft rendezvous using convolutional neural networks. In *2018 IEEE Aerospace Conference*. IEEE, March 2018. doi:[10.1109/aero.2018.8396425](https://doi.org/10.1109/aero.2018.8396425).
- [8] Sumant Sharma and Simone D'Amico. Pose estimation for non-cooperative spacecraft rendezvous using neural networks. In *Proceedings of the AIAA/AAS Space Flight Mechanics Meeting*, 2019. URL <https://arxiv.org/abs/1906.09868v1>. report number: AAS 19-350.
- [9] Alex Kendall, Matthew Grimes, and Roberto Cipolla. PoseNet: A convolutional network for real-time 6-DOF camera relocalization. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2938–2946, Santiago, Chile, October 2015. IEEE. doi:[10.1109/iccv.2015.336](https://doi.org/10.1109/iccv.2015.336).
- [10] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2015. doi:[10.1109/cvpr.2015.7298594](https://doi.org/10.1109/cvpr.2015.7298594).
- [11] Sen Wang, Ronald Clark, Hongkai Wen, and Niki Trigoni. DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2017. doi:[10.1109/icra.2017.7989236](https://doi.org/10.1109/icra.2017.7989236).
- [12] Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Trevor Darrell, and Kate Saenko. Long-term recurrent convolutional networks for visual recognition and description. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2015. doi:[10.1109/cvpr.2015.7298878](https://doi.org/10.1109/cvpr.2015.7298878).
- [13] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. FlowNet: Learning optical flow with convolutional networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, October 2015.

doi:[10.1109/iccv.2015.316](https://doi.org/10.1109/iccv.2015.316).

- 705 [14] Andreas Geiger, Julius Ziegler, and Christoph Stiller. StereoScan: Dense 3D reconstruction in real-time. In *2011 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, June 2011. doi:[10.1109/ivs.2011.5940405](https://doi.org/10.1109/ivs.2011.5940405).
- [15] Mahdi Rad and Vincent Lepetit. BB8: A scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth. In *2017 IEEE International Conference on Computer Vision (ICCV)*, October 2017. doi:[10.1109/iccv.2017.413](https://doi.org/10.1109/iccv.2017.413).
- 710 [16] Richard Szeliski. *Computer Vision: Algorithms and Applications*. Springer Science & Business Media, 2010. doi:[10.1007/978-1-84882-935-0](https://doi.org/10.1007/978-1-84882-935-0).
- [17] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014. URL <https://arxiv.org/abs/1409.1556>.
- [18] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2nd edition, 2004. doi:[10.1017/CBO9780511811685](https://doi.org/10.1017/CBO9780511811685).
- 715 [19] Shubham Sonawani, Ryan Alimo, Renaud Detry, Daniel Jeong, Andrew Hess, and Heni Ben Amor. Assistive relative pose estimation for on-orbit assembly using convolutional neural networks. In *AIAA Scitech 2020 Forum*. American Institute of Aeronautics and Astronautics, January 2020. doi:[10.2514/6.2020-2096](https://doi.org/10.2514/6.2020-2096).
- [20] Bo Chen, Jiewei Cao, Alvaro Parra, and Tat-Jun Chin. Satellite pose estimation with deep landmark regression and nonlinear pose refinement. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. IEEE, October 2019. doi:[10.1109/iccvw.2019.00343](https://doi.org/10.1109/iccvw.2019.00343).
- 720 [21] Wenxiu Huan, Mingmin Liu, and Qinglei Hu. Pose estimation for non-cooperative spacecraft based on deep learning. In *2020 39th Chinese Control Conference (CCC)*, pages 3339–3343, 2020. doi:[10.23919/CCC50068.2020.9189253](https://doi.org/10.23919/CCC50068.2020.9189253).
- 725 [22] Charles Oestreich, Tae W. Lim, and Randy Broussard. On-orbit relative pose initialization via convolutional neural networks. In *AIAA Scitech 2020 Forum*. American Institute of Aeronautics and Astronautics, January 2020. doi:[10.2514/6.2020-0457](https://doi.org/10.2514/6.2020-0457).
- [23] Kiruki Cosmas and Asami Kenichi. Utilization of FPGA for onboard inference of landmark localization in CNN-based spacecraft pose estimation. *Aerospace*, 7(11), 2020. ISSN 2226-4310. doi:[10.3390/aerospace7110159](https://doi.org/10.3390/aerospace7110159).
- 730 [24] Daichi Hirano, Hiroki Kato, and Tatsuhiko Saito. Deep learning based pose estimation in space. In *Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS)*, 2018. URL [https://robotics.estec.esa.int/i-SAIRAS/isairas2018/Papers/Session%203c/4\\_isairas2018\\_deep\\_ver1-39-77-Hirano-Daichi.pdf](https://robotics.estec.esa.int/i-SAIRAS/isairas2018/Papers/Session%203c/4_isairas2018_deep_ver1-39-77-Hirano-Daichi.pdf).
- [25] Lena Downes, Ted J. Steiner, and Jonathan P. How. Deep learning crater detection for lunar terrain relative navigation. In *AIAA Scitech 2020 Forum*. American Institute of Aeronautics and Astronautics, January 2020. doi:[10.2514/6.2020-1838](https://doi.org/10.2514/6.2020-1838).
- 735 [26] Lena M. Downes, Ted J. Steiner, and Jonathan P. How. Lunar terrain relative navigation using a convolutional neural network for visual crater detection. In *2020 American Control Conference (ACC)*, pages 4448–4453, 2020. doi:[10.23919/ACC45564.2020.9147595](https://doi.org/10.23919/ACC45564.2020.9147595).
- 740 [27] Lorenzo Pasqualetto Cassinis, Robert Fonod, Eberhard Gill, Ingo Ahrns, and Jesus Gil Fernandez. CNN-based pose estimation system for close-proximity operations around uncooperative spacecraft. In *AIAA Scitech 2020*

*Forum*. American Institute of Aeronautics and Astronautics, January 2020. doi:10.2514/6.2020-1457.

- [28] Tanner Campbell, Roberto Furfaro, Richard Linares, and David Gaylor. A deep learning approach for optical autonomous planetary relative terrain navigation. In *27th AAS/AIAA Space Flight Mechanics Meeting, 2017*, pages 3293–3302. Univelt Inc., 2017. URL [http://arclab.mit.edu/wp-content/uploads/2018/10/2017\\_06.pdf](http://arclab.mit.edu/wp-content/uploads/2018/10/2017_06.pdf).
- [29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL <https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>.
- [30] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. doi:10.1109/CVPR.2009.5206848.
- [31] Staffan Persson, Per Bodin, Eberhard Gill, Jon Harr, and John Jörgensen. PRISMA - An autonomous formation flying mission. In *Small Satellite Systems and Services - The 4S Symposium*, 2006. URL <https://elib.dlr.de/46839/>.
- [32] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, June 2017. doi:10.1109/tpami.2016.2577031.
- [33] Sumant Sharma and Simone D’Amico. Neural network-based pose estimation for noncooperative spacecraft rendezvous. *IEEE Transactions on Aerospace and Electronic Systems*, 56(6):4638–4658, 2020. doi:10.1109/TAES.2020.2999148.
- [34] Wadim Kehl, Fabian Manhardt, Federico Tombari, Slobodan Ilic, and Nassir Navab. SSD-6D: Making RGB-based 3D detection and 6D pose estimation great again. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, October 2017. doi:10.1109/iccv.2017.169.
- [35] Sumant Sharma, Jacopo Ventura, and Simone D’Amico. Robust model-based monocular pose initialization for noncooperative spacecraft rendezvous. *Journal of Spacecraft and Rockets*, 55(6):1414–1429, nov 2018. doi:10.2514/1.a34124.
- [36] 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)*, pages 6007–6013, Paris, France, 2020. doi:10.1109/ICRA40945.2020.9197244.
- [37] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 770–778, 2016. doi:10.1109/cvpr.2016.90.
- [38] Lingqiao Liu, Lei Wang, and Xinwang Liu. In defense of soft-assignment coding. In *2011 International Conference on Computer Vision*, pages 2486–2493, 2011. doi:10.1109/iccv.2011.6126534.
- [39] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015. URL <http://proceedings.mlr.press/v37/ioffe15.html>.



- [40] Nathan Koenig and Andrew Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 3, pages 2149–2154, 2004. doi:[10.1109/iros.2004.1389727](https://doi.org/10.1109/iros.2004.1389727).
- [41] Ryohei Arakawa, Yuri Matsushita, Toshiya Hanada, Yasuhiro Yoshimura, and Shuji Nagasaki. Attitude estimation of space objects using imaging observations and deep learning. In *Advanced Maui Optical and Space Surveillance Technologies Conference*, page 21, 2019. URL <https://amostech.com/TechnicalPapers/2019/Non-Resolved-Object-Characterization/Arakawa.pdf>.
- [42] Thaweerath Phisannupawong, Patcharin Kamsing, Peerapong Torteeka, Sittiporn Channumsin, Utane Sawangwit, Warunyu Hematulin, Tanatthep Jarawan, Thanaporn Somjit, Soemsak Yooyen, Daniel Delahaye, and Pisit Boonsrimuang. Vision-based spacecraft pose estimation via a deep convolutional neural network for noncooperative docking operations. *Aerospace*, 7(9):126, 2020. doi:[10.3390/aerospace7090126](https://doi.org/10.3390/aerospace7090126).
- [43] Odysseas Kechagias-Stamatis, Nabil Aouf, Vincent Dubanchet, and Mark A Richardson. DeepLO: Multi-projection deep LIDAR odometry for space orbital robotics rendezvous relative navigation. *Acta Astronautica*, 177:270–285, 2020. doi:[10.1016/j.actaastro.2020.07.034](https://doi.org/10.1016/j.actaastro.2020.07.034).
- [44] P.J. Besl and Neil D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, February 1992. doi:[10.1109/34.121791](https://doi.org/10.1109/34.121791).
- [45] Torsten Sattler, Qunjie Zhou, Marc Pollefeys, and Laura Leal-Taixe. Understanding the limitations of CNN-based absolute camera pose regression. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3302–3312, 2019. doi:[10.1109/cvpr.2019.00342](https://doi.org/10.1109/cvpr.2019.00342).
- [46] Tae Ha Park, Sumant Sharma, and Simone D’Amico. Towards robust learning-based pose estimation of noncooperative spacecraft. 2019. URL <http://arxiv.org/abs/1909.00392>.
- [47] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. 2018. URL <http://arxiv.org/abs/1804.02767>.
- [48] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted residuals and linear bottlenecks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2018. doi:[10.1109/cvpr.2018.00474](https://doi.org/10.1109/cvpr.2018.00474).
- [49] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. MobileNets: Efficient convolutional neural networks for mobile vision applications. 2017. URL <http://arxiv.org/abs/1704.04861>.
- [50] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. 2018. URL <http://arxiv.org/abs/1811.12231>.
- [51] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1501–1510, 2017. doi:[10.1109/iccv.2017.167](https://doi.org/10.1109/iccv.2017.167).
- [52] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2019. doi:[10.1109/cvpr.2019.00584](https://doi.org/10.1109/cvpr.2019.00584).



- [53] Yurong Huo, Zhi Li, and Feng Zhang. Fast and accurate spacecraft pose estimation from single shot space imagery using box reliability and keypoints existence judgments. *IEEE Access*, 8:216283–216297, 2020. doi:[10.1109/ACCESS.2020.3041415](https://doi.org/10.1109/ACCESS.2020.3041415).
- [54] Luis Ferraz, Xavier Binefa, and Francesc Moreno-Noguer. Leveraging feature uncertainty in the PnP problem. In *Proceedings of the British Machine Vision Conference 2014*. British Machine Vision Association, 2014. doi:[10.5244/c.28.83](https://doi.org/10.5244/c.28.83). URL <https://doi.org/10.5244%2Fc.28.83>.
- [55] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander Alemi. Inception-v4, Inception-ResNet and the impact of residual connections on learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2017. URL <http://arxiv.org/abs/1602.07261>.
- [56] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Computer Vision – ECCV 2014*, pages 740–755, 2014. doi:[10.1007/978-3-319-10602-1\\_48](https://doi.org/10.1007/978-3-319-10602-1_48).
- [57] Yang Ming. Multi-pattern 3d intelligent reconstruction method for non-cooperative space targets based on deep learning. Master’s thesis, Harbin Institute of Technology, Harbin, 2018.
- [58] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678, 2014. doi:[10.1145/2647868.2654889](https://doi.org/10.1145/2647868.2654889).
- [59] Lin Yi. Research on method of docking ring spatial position intelligent perception. Master’s thesis, Harbin Institute of Technology, Harbin, 2020. URL <https://cdmd.cnki.com.cn/Article/CDMD-10213-1020396416.htm>.
- [60] Chen Jiaming. Research and simulation of satellite positioning error compensation technology based on convolution neural network. Master’s thesis, Beijing University of Posts and Telecommunications, Beijing, 2018.
- [61] Hao Wang, Jie Jiang, and Guangjun Zhang. CraterIDNet: An end-to-end fully convolutional neural network for crater detection and identification in remotely sensed planetary images. *Remote sensing*, 10(7):1067, 2018. doi:[10.3390/rs10071067](https://doi.org/10.3390/rs10071067).
- [62] David Vaniman, Robert Reedy, Grant Heiken, Gary Olhoeft, and Wendell Mendell. The lunar environment. *The lunar Sourcebook*, pages 27–60, 1991. URL <https://ci.nii.ac.jp/naid/10003734174/en/>.
- [63] Ebrahim Emami, George Bebis, Ara Nefian, and Terry Fong. Automatic crater detection using convex grouping and convolutional neural networks. In *International symposium on visual computing*, pages 213–224, 2015. doi:[10.1007/978-3-319-27863-6\\_20](https://doi.org/10.1007/978-3-319-27863-6_20).
- [64] Joseph Paul Cohen, Henry Z Lo, Tingting Lu, and Wei Ding. Crater detection via convolutional neural networks. 2016. URL <http://arxiv.org/abs/1601.00978>.
- [65] Leon F Palafox, Christopher W Hamilton, Stephen P Scheidt, and Alexander M Alvarez. Automated detection of geological landforms on Mars using convolutional neural networks. *Computers & Geosciences*, 101:48–56, 2017. doi:[10.1016/j.cageo.2016.12.015](https://doi.org/10.1016/j.cageo.2016.12.015).
- [66] J. D. O’Keefe and T. J. Ahrens. Complex craters: Relationship of stratigraphy and rings to impact conditions. *Journal of Geophysical Research Planets*, 104:27091–27104, 1999. doi:[10.1029/1998je000596](https://doi.org/10.1029/1998je000596).
- [67] Michael R Klear. Pycda: An open-source library for automated crater detection. *Proceedings of the 9th Plane-*

tary Crater Consort, Boulder, CO, 2018. URL <http://planetarycraterconsortium.nau.edu/KlearPCC9.pdf>.

- [68] Stuart J. Robbins and Brian M. Hynek. A new global database of mars impact craters  $\geq 1$  km: 2. global crater properties and regional variations of the simple-to-complex transition diameter. *Journal of Geophysical Research: Planets*, 117(E6), June 2012. doi:[10.1029/2011je003967](https://doi.org/10.1029/2011je003967).
- 860 [69] Ari Silburt, Mohamad Ali-Dib, Chenchong Zhu, Alan Jackson, Diana Valencia, Yevgeni Kissin, Daniel Tamayo, and Kristen Menou. Lunar crater identification via deep learning. *Icarus*, 317:27–38, 2019. doi:[10.1016/j.icarus.2018.06.022](https://doi.org/10.1016/j.icarus.2018.06.022).
- [70] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Lecture Notes in Computer Science*, pages 234–241. Springer International Publishing, 2015.
- 865 doi:[10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28).
- [71] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1): 1929–1958, 2014. URL <http://www.cs.toronto.edu/~rsalakhu/papers/srivastava14a.pdf>.
- [72] RZ Povilaitis, MS Robinson, CH Van der Bogert, Harald Hiesinger, HM Meyer, and LR Ostrach. Crater density differences: Exploring regional resurfacing, secondary crater populations, and crater saturation equilibrium on
- 870 the moon. *Planetary and Space Science*, 162:41–51, 2018. doi:[10.1016/j.pss.2017.05.006](https://doi.org/10.1016/j.pss.2017.05.006).
- [73] James W Head, Caleb I Fassett, Seth J Kadish, David E Smith, Maria T Zuber, Gregory A Neumann, and Erwan Mazarico. Global distribution of large lunar craters: Implications for resurfacing and impactor populations. *science*, 329(5998):1504–1507, 2010. doi:[10.1126/science.1195050](https://doi.org/10.1126/science.1195050).
- 875 [74] Hoonhee Lee, Han-Lim Choi, Dawoon Jung, and Sujin Choi. Deep neural network-based landmark selection method for optical navigation on lunar highlands. *IEEE Access*, 8:99010–99023, 2020. doi:[10.1109/ACCESS.2020.2996403](https://doi.org/10.1109/ACCESS.2020.2996403).
- [75] Allan R Klumpp. Apollo lunar descent guidance. *Automatica*, 10(2):133–146, 1974. doi:[10.1016/0005-1098\(74\)90019-3](https://doi.org/10.1016/0005-1098(74)90019-3).
- 880 [76] Chirold D Epp, Edward A Robertson, and Tye Brady. Autonomous landing and hazard avoidance technology (ALHAT). In *2008 IEEE Aerospace Conference*, pages 1–7, 2008. doi:[10.1109/aero.2008.4526297](https://doi.org/10.1109/aero.2008.4526297).
- [77] Roberto Furfaro, Wolfgang Fink, and Jeffrey S Kargel. Autonomous real-time landing site selection for Venus and Titan using evolutionary fuzzy cognitive maps. *Applied Soft Computing*, 12(12):3825–3839, 2012. doi:[10.1016/j.asoc.2012.01.014](https://doi.org/10.1016/j.asoc.2012.01.014).
- 885 [78] Daniel Maturana and Sebastian Scherer. 3D convolutional neural networks for landing zone detection from LiDAR. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3471–3478, 2015. doi:[10.1109/icra.2015.7139679](https://doi.org/10.1109/icra.2015.7139679).
- [79] Paolo Lunghi and Michèle Lavagna. Autonomous vision-based hazard map generator for planetary landing phases. In *65th International Astronautical Congress (IAC)*, pages 5103–5114, 2014. URL <http://hdl.handle.net/11311/861150>.
- 890 [80] Paolo Lunghi, Marco Ciarambino, and Michèle Lavagna. A multilayer perceptron hazard detector for vision-based autonomous planetary landing. *Advances in Space Research*, 58(1):131–144, July 2016. doi:[10.1016/j.asr.2016.04.012](https://doi.org/10.1016/j.asr.2016.04.012).

- [81] Rahul Moghe and Renato Zanetti. On-line hazard detection algorithm for precision lunar landing using semantic segmentation. In *AIAA Scitech 2020 Forum*. American Institute of Aeronautics and Astronautics, January 2020. doi:[10.2514/6.2020-0462](https://doi.org/10.2514/6.2020-0462).
- [82] Rahul Moghe and Renato Zanetti. A deep learning approach to hazard detection for autonomous lunar landing. *The Journal of the Astronautical Sciences*, 67(4):1811–1830, 2020. doi:[10.1007/s40295-020-00239-8](https://doi.org/10.1007/s40295-020-00239-8).
- [83] Alexander Buslaev, Vladimir I Iglovikov, Eugene Khvedchenya, Alex Parinov, Mikhail Druzhinin, and Alexandr A Kalinin. Albumentations: Fast and flexible image augmentations. *Information*, 11(2):125, 2020. doi:[10.3390/info11020125](https://doi.org/10.3390/info11020125).
- [84] Leena Singh and Sungyung Lim. On lunar on-orbit vision-based navigation: Terrain mapping, feature tracking driven EKF. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, page 6834, 2008. doi:[10.2514/6.2008-6834](https://doi.org/10.2514/6.2008-6834).
- [85] Chengchao Bai, Jifeng Guo, Linli Guo, and Junlin Song. Deep multi-layer perception based terrain classification for planetary exploration rovers. *Sensors*, 19(14):3102, 2019. doi:[10.3390/s19143102](https://doi.org/10.3390/s19143102).
- [86] Bai Chengchao. *Research on vision/vibration based Terrain perception for rovers*. PhD thesis, Harbin Institute of Technology, Harbin, 2019.
- [87] Roberto Furfaro, Ilaria Bloise, Marcello Orlandelli, Pierluigi Di Lizia, Francesco Topputo, Richard Linares, et al. Deep learning for autonomous lunar landing. In *2018 AAS/AIAA Astrodynamics Specialist Conference*, pages 3285–3306, 2018. URL <http://hdl.handle.net/11311/1063150>.
- [88] Michael A. Patterson and Anil V. Rao. GPOPS-II: A MATLAB software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming. *ACM Transactions on Mathematical Software*, 41(1):1–37, October 2014. doi:[10.1145/2558904](https://doi.org/10.1145/2558904).
- [89] Roberto Furfaro and Andrew M. Law. Relative optical navigation around small bodies via extreme learning machines. In *2015 AAS/AIAA Astrodynamics Specialist Conference*, volume 156, pages 1959–1978, 2016. URL <http://www.scopus.com/inward/record.url?scp=85007336458&partnerID=8YFLogxK>.
- [90] Ari Silburt, Chenchong Zhu, Mohamad Ali-Dib, Kristen Menou, and Alan Jackson. DeepMoon: Convolutional neural network trainer to identify Moon craters. *Astrophysics Source Code Library*, 2018. URL <https://ui.adsabs.harvard.edu/abs/2018ascl.soft05029S>.
- [91] Mattia Pugliatti and Francesco Topputo. Navigation about irregular bodies through segmentation maps. In *31st Space Flight Mechanics Meeting*, pages AAS21–383, 2021. URL <http://hdl.handle.net/11311/1163932>.
- [92] P. M. Beauchamp, J. A. Cutts, C. Mercer, and L. A. Dudzinski. Technology Planning for NASA’s Future Planetary Science Missions. In *Planetary Science Vision 2050 Workshop*, volume 1989, page 8051, February 2017. URL <https://ui.adsabs.harvard.edu/abs/2017LPICo1989.8051B>.
- [93] Li Shuang and Cui Pingyuan. Landmark tracking based autonomous navigation schemes for landing spacecraft on asteroids. *Acta Astronautica*, 62(6-7):391–403, 2008. doi:[10.1016/j.actaastro.2007.11.009](https://doi.org/10.1016/j.actaastro.2007.11.009).
- [94] Tommaso Guffanti. Multi-objective autonomous spacecraft motion planning around near-earth asteroids using machine learning. Technical Report 1-6, 2018. URL <http://cs229.stanford.edu/proj2018/report/223.pdf>. CS 229: Final Project.
- [95] Khilan Ravani, S. Mathavaraj, and Radhakant Padhi. Site detection for autonomous soft-landing on aster-

oids using deep learning. *Transactions of the Indian National Academy of Engineering*, 6(2):365–375, 2021. doi:[10.1007/s41403-021-00207-0](https://doi.org/10.1007/s41403-021-00207-0).

- [96] Nathan Harl, Karthikeyan Rajagopal, and SN Balakrishnan. Neural network based modified state observer for orbit uncertainty estimation. *Journal of Guidance, Control, and Dynamics*, 36(4):1194–1209, 2013. doi:[10.2514/1.55711](https://doi.org/10.2514/1.55711).

- [97] Yu Song, Lin Cheng, and Shengping Gong. Fast estimation of gravitational field of irregular asteroids based on deep neural network and its application. In *Advances in the Astronautical Sciences AAS/AIAA Spaceflight Mechanics*, volume 168, pages AAS 19–397, 2019. URL [https://www.researchgate.net/profile/Yu-Song-45/publication/331523846\\_FAST\\_ESTIMATION\\_OF\\_GRAVITATIONAL\\_FIELD\\_OF\\_IRREGULAR\\_ASTEROIDS\\_BASED\\_ON\\_DEEP\\_NEURAL\\_NETWORK\\_AND\\_ITS\\_APPLICATION/links/5c7e5d9c458515831f8421a7/FAST-ESTIMATION-OF-GRAVITATIONAL-FIELD-OF-IRREGULAR-ASTEROIDS-BASED-ON-DEEP-NEURAL-NETWORK-AND-ITS-APPLICATION.pdf](https://www.researchgate.net/profile/Yu-Song-45/publication/331523846_FAST_ESTIMATION_OF_GRAVITATIONAL_FIELD_OF_IRREGULAR_ASTEROIDS_BASED_ON_DEEP_NEURAL_NETWORK_AND_ITS_APPLICATION/links/5c7e5d9c458515831f8421a7/FAST-ESTIMATION-OF-GRAVITATIONAL-FIELD-OF-IRREGULAR-ASTEROIDS-BASED-ON-DEEP-NEURAL-NETWORK-AND-ITS-APPLICATION.pdf).

- [98] Himangshu Kalita, Erik Asphaug, Stephen Schwartz, and Jekanthan Thangavelautham. Network of nano-landers for in-situ characterization of asteroid impact studies. In *68th International Astronautical Congress (IAC)*, Adelaide, Australia, 2017. URL <http://arxiv.org/abs/1709.02885>.

- [99] Lorenzo Feruglio, Sabrina Corpino, and Daniele Calvi. Neural networks for event detection: an interplanetary CubeSat asteroid mission case study. In *AIAA SPACE 2016*. American Institute of Aeronautics and Astronautics, sep 2016. doi:[10.2514/6.2016-5615](https://doi.org/10.2514/6.2016-5615).

- [100] Giulia Viavattene and Matteo Ceriotti. Artificial neural network for preliminary multiple nea rendezvous mission using low thrust. In *70th International Astronautical Congress (IAC), Washington, DC, USA*, 2019. URL <http://eprints.gla.ac.uk/202035/1/202035.pdf>.

- [101] Mattia Pugliatti and Francesco Topputo. Small-body shape recognition with convolutional neural network and comparison with explicit features based method. In *2020 AAS/AIAA Astrodynamics Specialist Conference*, pages 1–20, 2020. URL <https://re.public.polimi.it/retrieve/handle/11311/1145538/537728/PUGLM01-20.pdf>.

- [102] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. 2016. URL <http://arxiv.org/abs/1603.04467>.

- [103] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. URL [https://openaccess.thecvf.com/content\\_cvpr\\_2015/papers/Long\\_Fully\\_Convolutional\\_Networks\\_2015\\_CVPR\\_paper.pdf](https://openaccess.thecvf.com/content_cvpr_2015/papers/Long_Fully_Convolutional_Networks_2015_CVPR_paper.pdf).