

EFFICIENT FEATURE DESCRIPTION FOR SMALL BODY RELATIVE NAVIGATION USING BINARY CONVOLUTIONAL NEURAL NETWORKS

Travis Driver* and Panagiotis Tsiotras[†]

Missions to small celestial bodies rely heavily on optical feature tracking for characterization of and relative navigation around the target body. While techniques for feature tracking based on deep learning are a promising alternative to current human-in-the-loop processes, designing deep architectures that can operate onboard spacecraft is challenging due to onboard computational and memory constraints. This paper introduces a novel deep local feature description architecture that leverages binary convolutional neural network layers to significantly reduce computational and memory requirements. We train and test our model on real images of small bodies from legacy and ongoing missions and demonstrate increased performance relative to traditional handcrafted methods.

INTRODUCTION

Feature tracking is an integral component of current small body shape reconstruction and relative navigation approaches. The current state-of-the-practice for small body feature tracking leverages high-fidelity digital terrain maps (DTMs) of salient surface regions as local feature representations, which require extensive human involvement and mission operations planning for accurate construction.^{1,2} Autonomous keypoint detection and feature description methods based on deep convolutional neural networks (CNNs) are a promising alternative to the current human-in-the-loop approach,³ but the inherent computational overhead of deep architectures makes their implementation onboard spacecraft challenging.

Binary convolutional neural networks (BNNs) are a promising technology for enabling deep learning onboard future robotic spacecraft. BNNs represent weights and activations with 1 bit and use bitwise operations to perform cross-correlation, leading to significant computational, memory, and energy savings. Indeed, binary convolutions have been shown to be almost $60\times$ faster while using less than 1% of the energy compared to 32-bit convolutions on FPGAs and ASICs.⁴ While BNNs have traditionally led to significant performance degradation, recent work has demonstrated impressive performance on image classification by maximizing information entropy and minimizing quantization error within the BNN layers.⁵ Therefore, we propose a novel and efficient feature description architecture with BNN layers for deep feature tracking onboard spacecraft.

*PhD Student, Institute for Robotics and Intelligent Machines, Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30313, USA.

[†]David and Lewis Chair and Professor, Institute for Robotics and Intelligent Machines, Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30313, USA.

PROPOSED APPROACH

We propose a novel local feature description architecture that leverages BNN layers to efficiently and reliably describe extracted image patches. Our model is trained on real imagery acquired during legacy and ongoing missions to small bodies.

Binary Convolutional Neural Networks

The main operation in CNNs can be formulated as

$$z = \mathbf{w}^\top \mathbf{a}, \quad (1)$$

where $\mathbf{w}$ and $\mathbf{a}$ are the weight tensor and the input activation tensor from the previous layer, respectively. BNNs represent the weights and activations as 1-bit binary values for more efficient computation and storage. In general, binary network quantization can be formulated as

$$z = Q_w(\mathbf{w})^\top Q_a(\mathbf{a}) = \alpha\beta(\mathbf{b}_w \odot \mathbf{b}_a), \quad (2)$$

where $\mathbf{b}_w, \mathbf{b}_a \in \{-1, +1\}^d$ are the quantized 1-bit weights and activations, respectively, with corresponding scalars α and β , and $\odot$ denotes the inner product with bitwise operations `XNOR` and `bitcount`. The quantization function Q is typically taken to be the `sign` function:

$$Q(\mathbf{x}) = \begin{cases} +1 & \text{if } \mathbf{x} \geq 0 \\ -1 & \text{otherwise} \end{cases}. \quad (3)$$

We leverage IR-Net⁵ layers for binary quantization, which implement a stochastic quantization method to reduce information loss and quantization error from the forward `sign` operator and an incremental function approximation for the backward gradient. Specifically, the stochastic binarization method models each element b in the quantized weights $\mathbf{b}_w$ as a Bernoulli random variable with parameter p :

$$f(b; p) = \begin{cases} p & \text{if } b = +1 \\ 1 - p & \text{if } b = -1 \end{cases}. \quad (4)$$

Therefore, the information entropy of the quantized elements is maximized with $p = 0.5$, which is enforced via weight standardization:⁵

$$\hat{\mathbf{w}} = \frac{\mathbf{w} - \mu}{\sigma}, \quad (5)$$

where μ and σ are the mean and standard deviation of the filter weights $\mathbf{w}$. Additionally, an integer bit-shift value $s \in \mathbb{Z}$ is introduced to minimize the quantization error between the full-precision and binary weights, i.e.,

$$s^* := \arg \min_s \|\hat{\mathbf{w}} - \tilde{Q}_w(\hat{\mathbf{w}}) \lll s \|^2, \quad (6)$$

where $\lll$ is the left ($s > 0$) or right ($s < 0$) bit-shift operator. Taking $\tilde{Q}_{\hat{\mathbf{w}}}(\mathbf{w}) := \text{sign}(\hat{\mathbf{w}})$ gives⁵

$$s^* = \text{round}(\log_2(\|\hat{\mathbf{w}}\|_1 / \dim(\hat{\mathbf{w}}))). \quad (7)$$

Finally, weight and activation binarization for the forward step is given by

$$\begin{aligned} \mathbf{b}_w &= Q_w(\hat{\mathbf{w}}) = \text{sign}(\mathbf{w}) \lll s, \\ \mathbf{b}_a &= Q_a(\mathbf{a}) = \text{sign}(\mathbf{a}), \end{aligned}$$

resulting in activations z of the form

$$z = (\mathbf{b}_w \odot \mathbf{b}_a) \lll s. \quad (8)$$

Next, the discontinuity due to the `sign` operation in the forward step is approximated during back-propagation by⁵

$$g_{t,k}(x) = k \tanh(tx), \quad (9)$$

where k and t are control variables that are incrementally adjusted during the training process:

$$t := T_{\min} 10^{\frac{i}{n} \log(T_{\max}/T_{\min})}, \quad k := \max(1/t, 1), \quad (10)$$

where i is the current epoch, n is the total number of epochs, and $T_{\min}$ and $T_{\max}$ are taken to be 0.1 and 10, respectively.

Network Architecture

Keypoint detection and feature description methods operate by computing *keypoints* which localize salient regions in the image and *descriptors* which form a latent representation of the local image data at detected keypoints, where the descriptor typically takes the form of a d -dimensional vector. Feature tracking is conducted by matching features between frames with respect to the Euclidean distance between their corresponding descriptors. We leverage the *detect-then-describe* paradigm whereby patches are first extracted using a handcrafted keypoint detection scheme and then described by passing the patches through the description network. This is in contrast to the *detect-and-describe* paradigm whereby the network takes in the full image and outputs dense saliency and descriptor maps that cover the entire image.³ The detect-then-describe paradigm is leveraged for two main reasons: description is only conducted on regions centered around detected keypoints, thus reducing unnecessary computations, and *multiscale* detection can be conducted without performing the expensive description step.

We leverage the detection scheme of SIFT⁶ for patch extraction. SIFT keypoints are extracted from local extrema of the saliency map derived by convolving the difference of Gaussians (DoG) kernel with the input image, as the DoG function provides a close approximation to the scale-normalized Laplacian of Gaussian function which has been shown to be scale invariant.⁷ We also plan on experimenting with other more efficient detection methods such as FAST⁸ and CenSurE.⁹ We use HyNet¹⁰ as our base description architecture, which consists of seven convolutional layers and outputs d -dimensional descriptors. Each convolutional layer except the last one before the final Batch Normalisation (BN) layer is followed by Filter Response Normalization (FRN) and a Thresholded Linear Unit (TLU)¹¹ activation layer. We replace all full-precision CNN layers except the first and last with the binary IR-Net layers. The full architecture is illustrated in Figure 1.

Moreover, the hybrid loss proposed by Tian et al.¹⁰ is used to learn discriminative local feature descriptors. Consider correspondences $\mathcal{M} := \{(k, \tau(k)) \mid \tau : K \leftrightarrow K'\}$ between descriptors $\{\mathbf{d}_k\}_{k \in K}$ and $\{\mathbf{d}'_{k'}\}_{k' \in K'}$. First, a regularization term is used to account for varying descriptor magnitudes before the final L_2 normalisation step:

$$R_{L_2} = \frac{1}{|\mathcal{M}|} \sum_{(k,k') \in \mathcal{M}} (\|\mathbf{x}_k\| - \|\mathbf{x}'_{k'}\|)^2, \quad (11)$$

where $\mathbf{x}_k$ and $\mathbf{x}'_{k'}$ are the corresponding descriptors *before* the final normalization step. Next, a triplet margin ranking loss that considers both the inner product and L_2 distance between descriptors

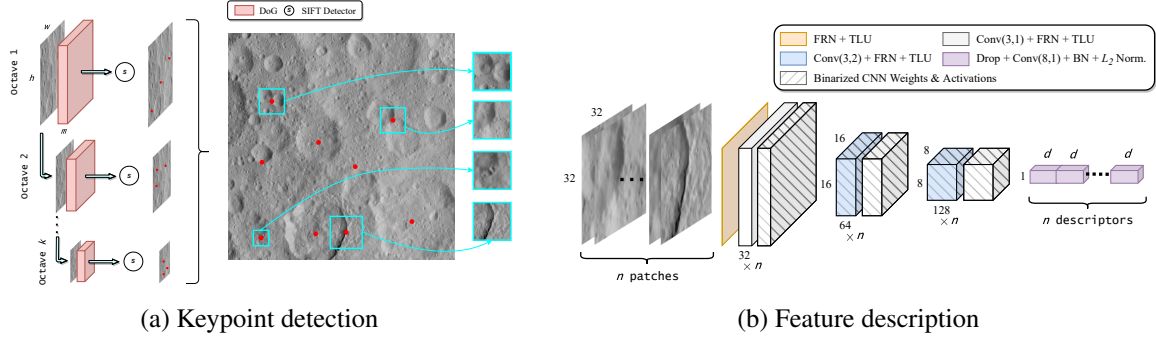


Figure 1: Feature extraction architecture.

is used, which has been shown to allow for more balanced gradient updates between positive and negative pairs:¹⁰

$$\mathcal{L}_{\text{triplet}} = \frac{1}{|\mathcal{M}|} \sum_{(k,k') \in \mathcal{M}} \max(0, m + s_H(\mathbf{d}_k, \mathbf{d}'_{k'}) - s_H(\mathbf{d}_k^-, \mathbf{d}'_{k'}^-)), \quad (12)$$

$$s_H(\mathbf{d}, \mathbf{d}') = \frac{1}{Z} (\alpha(1 - \mathbf{d}^\top \mathbf{d}') + \|\mathbf{d} - \mathbf{d}'\|), \quad (13)$$

where $\mathbf{d}_k^-, \mathbf{d}'_{k'}^-$ is the hardest negative sample for the pair $\mathbf{d}_k, \mathbf{d}'_{k'}$, i.e.,

$$\mathbf{d}_k^-, \mathbf{d}'_{k'}^- = \begin{cases} \mathbf{d}_k^\vee, \mathbf{d}'_{k'}^\vee & \text{if } \|\mathbf{d}_k^\vee - \mathbf{d}'_{k'}^\vee\| < \|\mathbf{d}_k - \mathbf{d}'_{k'}^\vee\| \\ \mathbf{d}_k, \mathbf{d}'_{k'}^\vee & \text{otherwise} \end{cases}, \quad (14)$$

$$\mathbf{d}_k^\vee := \arg \min_{\mathbf{d}_l, l \neq k} \|\mathbf{d}_l - \mathbf{d}'_{k'}^\vee\|, \quad \mathbf{d}'_{k'}^\vee := \arg \min_{\mathbf{d}'_{l'}, l' \neq k'} \|\mathbf{d}_k - \mathbf{d}'_{l'}\|, \quad (15)$$

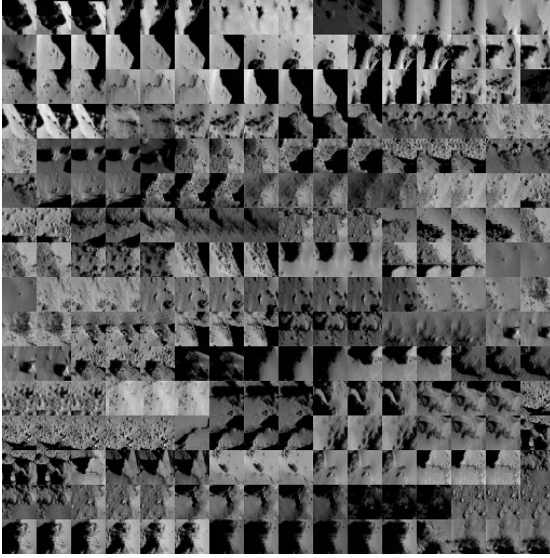
m is the margin between the positive and negative samples, $\alpha \in \mathbb{R}^+$ adjusts the ratio between s and d , and Z ensures the gradient has a maximum magnitude of 1. The hyperparameters m and α are empirically set to 1.2 and 2, respectively. Finally, the overall loss function is defined as

$$\mathcal{L} = \mathcal{L}_{\text{triplet}} + \gamma R_{L_2}, \quad (16)$$

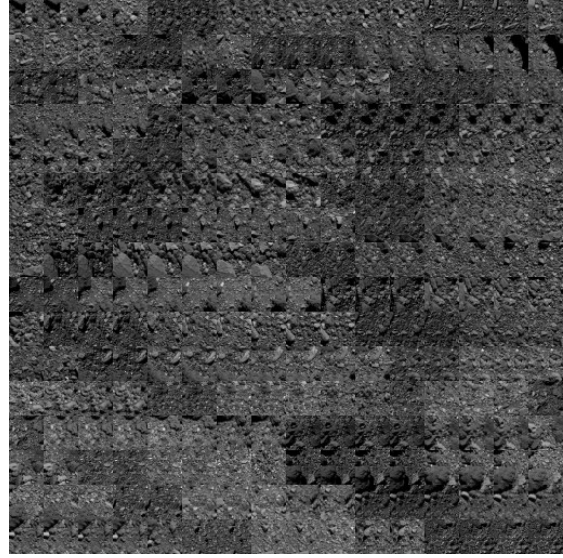
where γ is a regularization parameter which we set to $\gamma = 0.1$.

Dataset Generation

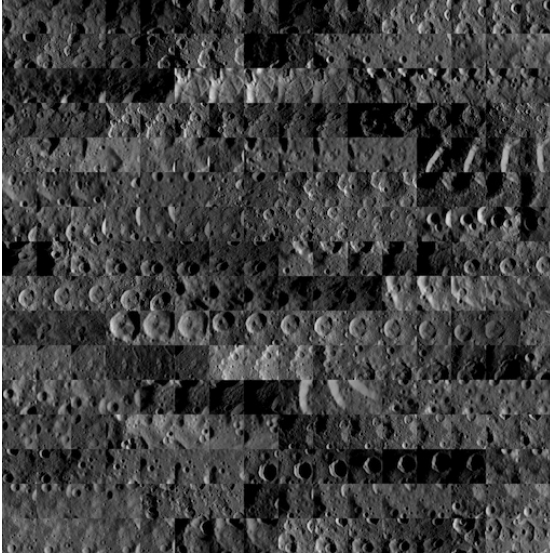
We leverage the AstroVision dataset³ to build a patch dataset for training and testing. Specifically, SIFT keypoints are extracted from randomly sampled image pairs and ground truth correspondences are estimated by projecting each keypoint from one image to the other using the ground truth poses and depth maps. A correspondence is taken to be valid if the projected keypoint falls within 3 pixels of the matched keypoint in the other image, and a bijective check is performed to ensure one-to-one correspondences, as in Ebel et al.¹² Finally, patches are extracted by sampling a square region of length 16η around each keypoint, where η is the SIFT scale estimate. This process was conducted for the Dawn @ 1 Ceres, Dawn @ 4 Vesta, OSIRIS-REx @ 101955 Bennu, and Rosetta @ 67P AstroVision datasets to build corresponding patch datasets. Patches from each of the datasets are provided in Figure 2.



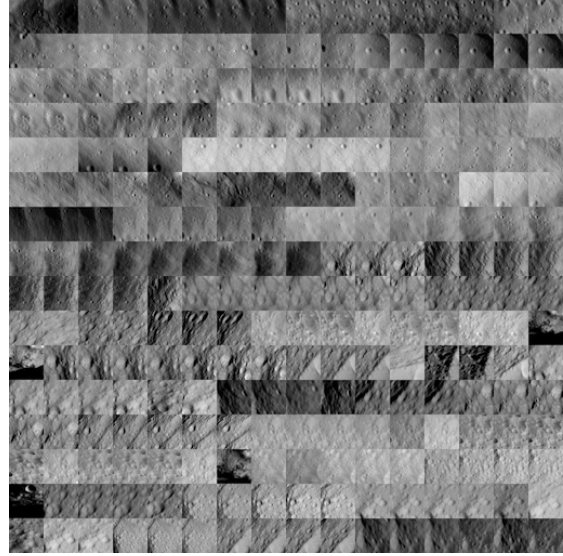
Rosetta @ 67P



OSIRIS-REx @ 101955 Bennu



Dawn @ 1 Ceres



Dawn @ 4 Vesta

Figure 2: Example patch mosaics.

RESULTS

As in previous works on feature description,^{10,12,13} we evaluate the performance of our network on patch matching using the popular false positive rate at 95% true positive recall (FPR95) metric, i.e., the ratio of negative descriptor pairs whose distance is within the distance threshold for 95% recall of the true positive pairs. Therefore, a low FPR95 value indicates that the network is able to reliably discriminate between positive and negative descriptor pairs, which is essential for robust matching between images. We also provide the number of millions of floating point operations

(MFLOPs) for computing a single patch descriptor as a metric for computational efficiency*. The network is trained on one dataset and tested on the remaining three, as in the seminal UBC benchmark.¹³ The results provided in Table 1 demonstrate that the network with BNN layers, referred to as HyNet-B, is able to significantly outperform the handcrafted descriptors computed by SIFT, as the HyNet-B descriptors achieve a significantly lower FPR95 value. Moreover, HyNet-B achieves comparable performance to its full-precision counterpart while requiring $\sim 20\times$ fewer FLOPs. We also modified the network to output a *binary* descriptor of length 256, denoted by $d = 256, 1/0$ in Table 1, which allows for efficient matching via the Hamming distance.¹⁶ The learned binary descriptors outperform the full-precision descriptors of SIFT while having a significantly smaller memory footprint.

Table 1: Patch matching performance on the AstroVision dataset. Performance is measured with respect to the false positive rate at 95% true positive recall (FPR95). We also provide the number of millions of floating point operations (MFLOPs) for computing a single patch descriptor as a metric for computational efficiency.

Train	Bennu	Ceres	Vesta	67P	Ceres	Vesta	67P	Bennu	Vesta	67P	Bennu	Ceres	
Test	67P			Bennu			Ceres			Vesta			MFLOPs
SIFT ($d = 128$)		71.6			76.83			69.29			73.26		—
HyNet ($d = 128$)	1.50	0.78	0.70	20.00	15.54	15.37	0.21	0.31	0.01	0.43	1.49	0.01	39.67
HyNet-B													
$d = 128$	3.70	2.65	2.14	34.47	27.26	30.78	1.40	1.89	0.04	5.25	9.55	0.98	2.51
$d = 256, 1/0$	8.96	6.37	6.30	47.76	38.99	46.56	5.97	3.92	1.00	6.44	16.32	6.57	3.56

CONCLUSION

In this paper, we presented a novel deep local feature description architecture that leverages BNN layers to significantly reduce the computational requirements of the model. The model was validated on *real* remote imagery of small bodies and demonstrated significant performance increases with respect to handcrafted feature description methods. We believe our network could viably operate on-board spacecraft equipped with the next generation space processor,¹⁷ which is projected to handle 32 GFLOPs per second,¹⁸ well above the theoretical computational requirements of our model. Before the final manuscript submission, we plan on adding an experiment for image-to-image matching using the AstroVision³ benchmarking suite and implementing our models on flight-relevant hardware.

ACKNOWLEDGEMENTS

This work was supported by NASA Space Technology Graduate Research Opportunity award 80NSSC21K1265.

REFERENCES

- [1] O. Barnouin, M. Daly, E. Palmer, C. Johnson, R. Gaskell, M. Al Asad, E. Bierhaus, K. Craft, C. Ernst, R. Espiritu, H. Nair, G. Neumann, L. Nguyen, M. Nolan, E. Mazarico, M. Perry, L. Philpott, J. Roberts, R. Steele, J. Seabrook, H. Susorney, J. Weirich, and D. Lauretta, “Digital terrain mapping by the OSIRIS-REx mission,” *Planetary and Space Science*, Vol. 180, 2020, p. 104764.

*The ratio between binary operations (BOPs) and FLOPs is taken to be $1 \text{ BOP} = \frac{1}{64} \text{ FLOP}$.^{14,15}

- [2] E. E. Palmer, R. Gaskell, M. G. Daly, O. S. Barnouin, C. D. Adam, and D. S. Lauretta, "Practical Stereophotoclinometry for Modeling Shape and Topography on Planetary Missions," *Planetary Science*, Vol. 3, No. 102, 2022, pp. 1–16.
- [3] T. Driver, K. Skinner, M. Dor, and P. Tsiotras, "AstroVision: Towards Autonomous Feature Detection and Description for Missions to Small Bodies Using Deep Learning," *preprint*, 2022. arXiv:2208.02053.
- [4] A. Mishra, E. Nurvitadhi, J. J. Cook, and D. Marr, "WRPN: Wide Reduced-Precision Networks," *Int. Conf. on Learning Representations (ICLR)*, 2018, pp. 1–11.
- [5] H. Qin, R. Gong, X. Liu, M. Shen, Z. Wei, F. Yu, and J. Song, "Forward and backward information retention for accurate binary neural networks," *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2250–2259.
- [6] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Int. J. of Computer Vision (IJCV)*, Vol. 60, Nov. 2004, pp. 91–110.
- [7] T. Lindeberg, "Scale-space theory: A basic tool for analyzing structures at different scales," *J. of Applied Statistics*, Vol. 21, No. 1-2, 1994, pp. 225–270.
- [8] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," *European Conf. on Computer Vision (ECCV)*, Springer, 2006, pp. 430–443.
- [9] M. Agrawal, K. Konolige, and M. R. Blas, "CenSurE: Center surround extremas for realtime feature detection and matching," *European Conf. on Computer Vision (ECCV)*, Springer, 2008, pp. 102–115.
- [10] Y. Tian, A. Barroso Laguna, T. Ng, V. Balntas, and K. Mikolajczyk, "HyNet: Learning local descriptor with hybrid similarity measure and triplet loss," *Neural Information Processing Systems (NeurIPS)*, 2020, pp. 1–12.
- [11] S. Singh and S. Krishnan, "Filter response normalization layer: Eliminating batch dependence in the training of deep neural networks," *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11237–11246.
- [12] P. Ebel, A. Mishchuk, K. M. Yi, P. Fua, and E. Trulls, "Beyond cartesian representations for local descriptors," *IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, 2019, pp. 253–262.
- [13] M. Brown, G. Hua, and S. Winder, "Discriminative learning of local image descriptors," *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, Vol. 33, No. 1, 2010, pp. 43–57.
- [14] Z. Liu, B. Wu, W. Luo, X. Yang, W. Liu, and K.-T. Cheng, "Bi-Real Net: Enhancing the performance of 1-bit CNNs with improved representational capability and advanced training algorithm," *European Conf. on Computer Vision (ECCV)*, 2018, pp. 722–737.
- [15] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "XNOR-Net: ImageNet classification using binary convolutional neural networks," *European Conf. on Computer Vision (ECCV)*, Springer, 2016, pp. 525–542.
- [16] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," *IEEE Int. Conf. on Computer Vision (ICCV)*, 2011, pp. 2564–2571.
- [17] R. Doyle, R. Some, W. Powell, K. Avery, G. Mounce, M. Lowry, M. Johnson, L. Bergman, W. Whitaker, and M. Goforth, "Next generation space processor (NGSP) high performance spaceflight computing (HPSC) next steps at NASA and AFRL," tech. rep., Jet Propulsion Laboratory, NASA, 2013.
- [18] T. Claudet, K. Tomita, and K. Ho, "Benchmark Analysis of Semantic Segmentation Algorithms for Safe Planetary Landing Site Selection," *IEEE Access*, Vol. 10, 2022, pp. 41766–41775.