

Render-to-real image dataset and CNN pose estimation for down-link restricted spacecraft missions

Andrew Price
Tohoku University
6-6-1 Aoba, Aramaki, Aoba-ku
980-8597 Sendai, Miyagi, Japan
pricea@dc.tohoku.ac.jp

Til Reichelt
Technische Universität Dresden
Marschnerstraße 32
01307 Dresden, Germany
til.reichelt@mailbox.tu-dresden.de

Kentaro Uno
Tohoku University
6-6-1 Aoba, Aramaki, Aoba-ku
980-8597 Sendai, Miyagi, Japan
unoken@tohoku.ac.jp

Swapnil Parekh
International Space University
1 rue Jean-Dominique Cassini
67400 Illkirch-Graffenstaden, France
swapnil.parekh@community.isunet.edu

Kazuya Yoshida
Tohoku University
6-6-1 Aoba, Aramaki, Aoba-ku
980-8597 Sendai, Miyagi, Japan
yoshida.astro@tohoku.ac.jp

Abstract—In an environment of escalating usage of Low Earth Orbit, the active remediation of debris is an increasingly necessary capability. Computer vision pose estimation is a core competency of active debris remediation but state-of-the-art pose estimation methods continue to grow in size and complexity. For bandwidth limited and edge computing cases, smaller networks are more feasibly implemented. A 16,845 synthetic image dataset, applicable to the upcoming JAXA CRD2 project, is rendered and a small pose estimation network is constructed and trained on the dataset. The network is then quantized, reducing the memory requirement by a factor of 8x to a theoretical size of 5.5 MB. The 5.5 MB network demonstrates sufficient accuracy in both single image pose and motion prediction tasks when compared to the full precision 32 bit network.

A summary of this work can be seen in Figure 1.

TABLE OF CONTENTS

1. INTRODUCTION.....	1
2. RELATED WORK	2
3. DATASET	3
4. POSE ESTIMATION NETWORK	5
5. QUANTIZATION CASE STUDY	7
6. CONCLUSION	8
REFERENCES	9
BIOGRAPHY	11

1. INTRODUCTION

Interest in the development and use of satellite technologies is increasing. Historically within the domain of civil and defense agendas but now also fueled by commercial constellations, the occupancy of Low Earth Orbit (LEO) is expected to triple in the next couple of years [1]. A higher LEO object count raises the chance of in-orbit collisions which further increases the likelihood of generating additional LEO objects. Termed the Kessler syndrome, this phenomenon was originally modelled in 1978 and recommended various debris mitigation engineering decisions [2]. Today, the orbital debris problem is globally recognized [3] [4] [5] and, as shown in Figure 2, growing [6].

The remediation of existing debris is a key research and development thrust for the reduction of orbital debris [7] [8]

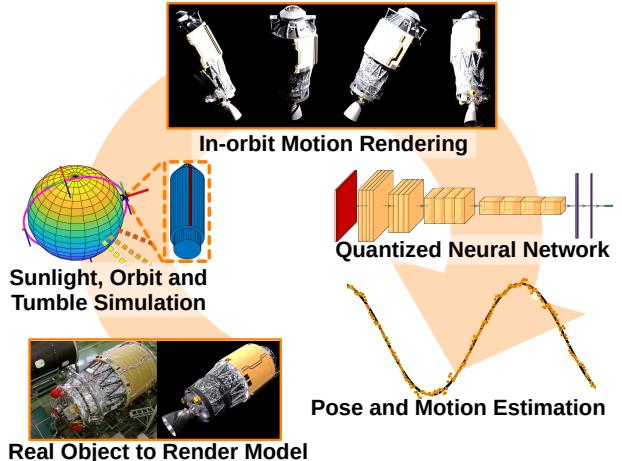


Figure 1: Conceptual summary illustration of this work.

[9]. The strategic Active Debris Removal (ADR) of certain key debris targets is a necessary measure for the stabilization of LEO [10].

Satellite Pose Estimation

Irrespective of mission specifics, the ability to **remotely** detect, track and analyze target inertial information is a fundamental technical capability for ADR [13]. However, the space sector faces a unique set of challenges compared to terrestrial applications.

One major limitation is the lack of relevant data. Training an image classifier or pose estimator for an object such as a car or human is a relatively well defined engineering problem; 100,000+ ImageNet [11] and 25,000+ MPII [12] are just two examples of the expansive datasets available for terrestrial classification and pose estimation tasks. For satellite pose estimation, datasets are much more limited. While rendering a satellite image is comparably simple, it remains challenging to acquire real satellite images in orbit. It is relatively straightforward to train or optimize a pose estimation system on rendered images, but transferring this pose estimation performance across the domain gap to real images remains a challenge.

A second major limitation is the minimal processing power

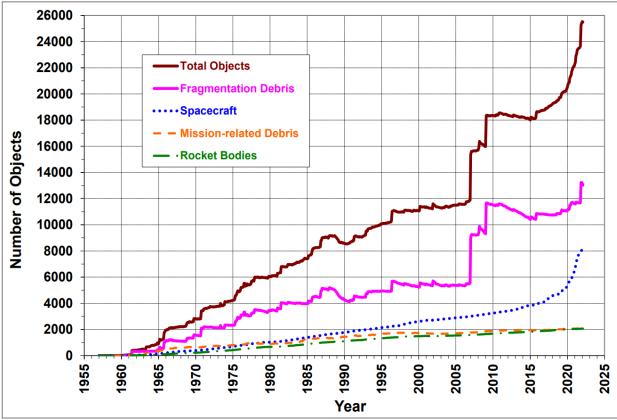


Figure 2: Historical catalogued objects in Earth orbits. [6]

for edge computing or minimal bandwidth for data transmission. Consider 3 examples:

- A **deep space mission** wherein the service spacecraft is only able to transmit back a handful of images.
- A time critical **orbiting mission** where the service spacecraft is only able to transmit images when in range of ground stations.
- An **autonomous mission** where the service spacecraft is attempting to perform pose estimation on a small on-board processor.

For each of these cases, a large consideration to be addressed is: which images to prioritize processing? Evidently the image selection process must be optimized to ensure suitable accuracy with limited bandwidth or a pose estimator must be optimized to operate on the spacecraft autonomously.

Objective

Currently, there are a number of ADR programs including RemoveDebris [19], Clearspace [20] and the JAXA Commercial Removal of Debris Demonstration (CRD2) [21]. It is in the context of the Japanese project that this ADR analysis is performed; this work contained three phases:

(I) Dataset

In preparation for the rocket body dataset, a render-to-real validation study was first performed to analyze the domain gap between synthetic renders and real images in the context of image RGB values. A physical micro-satellite model and a corresponding render model were created. Continuing previous work, physics and sensor-based noise were included to help shrink the domain gap. Based on lessons learned, focus then shifted to the mission specific dataset. A rocket body model dataset was rendered; model geometry and material properties were based on publicly available information for a Japanese H2A rocket body.

(II) Pose Estimation Network

In the context of hardware feasible implementation and simplicity, a CNN straight-shot pipeline was trained for pose estimation. The spacecraft motion was estimated from the CNN poses and the motion estimation error was compared for a number of different image sampling strategies to reduce image down-link requirements.

(III) Quantization Case Study

A short case study was completed for a quantized neural network using the Incremental Network Quantization (INQ) approach [22]. Pose and motion estimation results were then compared to the baseline network to investigate the feasibility of on-board satellite microprocessor pose estimation.

The original contributions of this paper may be identified as follows:

- A freely available H2A upper stage synthetic dataset. Although the dataset does not match the fidelity of state-of-the-art Spacecraft PosE Estimation Dataset (SPEED), the dataset is highly relevant to the current and active space debris removal mission: CRD2. Augmentation of this dataset is a trivial task and thus can easily accommodate more tumble motion simulations and different targets. Additionally the dataset contains motion sequences enabling both single image *pose* estimation and multiple image *motion* estimation.
- The quantization of a hardware-feasible spacecraft pose estimation network. This is a relatively new application of quantization and to the best of our knowledge is only beginning to be demonstrated [23].

2. RELATED WORK

Vision-based pose estimation for in-space applications did not begin with neural networks. The Synchronized Positional Hold Engage Reorient Experimental Satellites (SPHERES) was installed on the International Space Station (ISS) in 2006 [24] [25] and was upgraded to perform visual mapping and localization in 2012 [26] [27]. This platform contained two cameras for stereo-vision as well as a supporting assortment of gyros and ultrasonic sensors. The gyros supplemented the algorithm's rate estimations, the targets' surfaces were supplemented with high contrast patterned stickers and the experiment took place in the relatively uniform lighting environment of the ISS.

[28] used a monocular system and employed Gaussian process regression as a pose estimator. The tumble motion was restricted to 1D or 2D movement but the images were subject to different lighting conditions and noise; additionally this work was validated experimentally.

However, there is increasing support for neural network estimation approaches. Especially in support of the active debris remediation problem, a number of competitions and studies have been completed featuring deep learning.

The Satellite Pose Estimation Challenge [29] developed SPEED [15] using OpenGL synthetic images in addition to real images of the Tango spacecraft from the Prisma mission [30]. The competition produced a number of high accuracy pose estimation pipelines. The highest accuracy pipelines featured a keypoint detector Convolutional Neural Network (CNN) architecture paired with a Perspective-n-Point (PnP) algorithm to match the CNN output 2D correspondences to the target 3D correspondences in post-processing [31] [32]. There was additionally some high performing direct regression networks that did not make use of a post-processing PnP algorithm [33].

The SwissCube spacecraft pose estimation dataset improved synthetic image fidelity further by, in addition to physically realistic target rendering, including realistic modelling of the Sun, Earth and star backdrops [34]. A further contribution of this work was addressing the large scale variations charac-

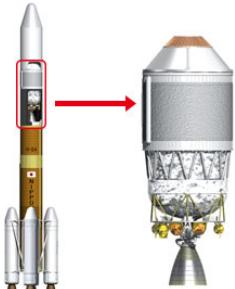


Figure 3: H2A 2nd stage.
[36]

teristic of space imagery (i.e. orbital debris can range from mm to m in size and m to km in distance from the camera). The proposed pose estimation pipeline used a feature pyramid network to regress 2D projections at multiple scales.

More recently, the Stanford Space Rendezvous Laboratory has developed the SPEED+ dataset [35], which includes additional hardware-in-the-loop images of a physical spacecraft mockup model; the intention of this dataset is to further augment the synthetic SPEED dataset with realistic images to enable pose estimation pipelines to better perform real image inferences after training on synthetic images.

Terrestrial Pose Estimation

Spacecraft pose estimation is a relatively niche application of pose estimation technology. Consequently, many applicable developments are first discovered and introduced in earth-bound research. The SwissCube spacecraft pose estimation dataset was developed from work on hyper realistic material rendering [16]. Their work introduced a novel parameterization of a material’s reflective properties, enabling the ability to produce hyper realistic renderings of objects using only material samples. This work is directly applicable to rendering hyper realistic images of objects in space that do not have prior images.

Spacecraft pose estimation pipeline architectures are typically adapted from terrestrial applications as well. The most successful 6D pose direct regression network from the Satellite Pose Estimation Challenge utilized an architecture similar in structure to PoseCNN. PoseCNN was originally developed for the pose estimation of household objects [17]. Recent 6D pose estimation accuracy improvements are trending in the direction of denser 2D to 3D correspondence matching as shown in ZebraPose [18]. While accurate, these denser correspondence representations often require a correspondingly heavier memory usage; however, a heavier memory requirement further lowers the feasibility of installing a pose estimation network on memory-limited spacecraft hardware.

3. DATASET

Based on publicly available data, the H2A rocket 2nd stage was used as the target, viewable in Figure 3.

Dataset Development Validation Study

Without access to a physical model of the desired H2A 2nd stage target, the objective was to create a render dataset. As stated previously, the state of the art hyper realistic spacecraft rendering, as demonstrated in the SwissCube dataset,

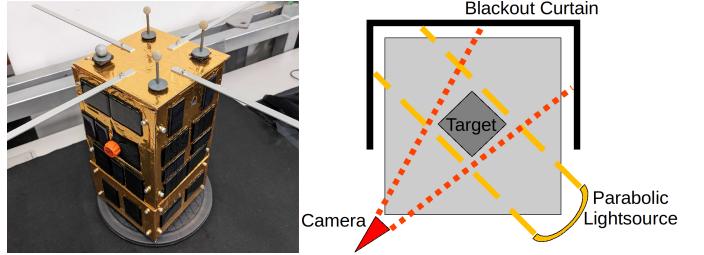


Figure 4: Micro-satellite physical model (left) and experimental setup (right).

required access to the reflective properties of the target materials [16]. Without access to the material properties or the means to obtain them, it was determined that a separate render-to-real analysis would be necessary to validate the render dataset construction process. This further created an opportunity to study the performance of pose estimation systems across the domain gap in future studies.

Micro-satellite physical model—The micro-satellite physical model was constructed as a 2x scale 2U satellite with dimensions 300 x 160 x 160 cm. The 2U physical model can be seen in Figure 4 (left). The surface was primarily composed of a typical satellite Multi-Layer Insulation (MLI) and two varieties of solar panels. Inside, an air tank fed air to 3 air-bearings located on the bottom of the model. Four optical tracking markers and four simple aluminum antennas were mounted on top. Additionally, two PCB cameras were recessed into the casing.

Images were taken in a dark room. The micro-satellite model rested on a granite testbed with three air bearings to enable minimal friction 2D planar motion. The table was surrounded by blackout curtains. The sole source of light was from a 1000W parabolic light source. The camera was aligned so as to avoid detecting primary light reflections from the background. A schematic of the setup can be viewed in Figure 4 (right).

Micro-satellite render model—The render model geometry was constructed in Solidworks and then imported to Blender. Blender’s Cycles renderer is a ray-trace engine with several shader options including physically based shading. Material properties were instead substituted by importing an image of the material as a colormap and iteratively modifying the blender Bidirectional Scattering Distribution Function (BSDF) until a suitable output image quality was obtained; the process flow is shown in Figure 5. A directional light source with an intensity of $1000 \frac{W}{m^2}$ was used [37].

Note that as a ray-tracer engine Blender supports self-reflections and self-occlusion. To best match real conditions the selection of lighting sources in Blender must be chosen with care. In orbit, direct sunlight, Earth albedo, Earth originated radiation and atmospheric refraction effects may all produce observable influences on an image. These effects were not modelled here; the objective was to reproduce the dark room experimental environment.

Render to Real Comparison—For comparison, a real experimental image and a render of the micro-satellite are shown in Figure 6. Immediately observable is the lack of complex bloom behaviour in the blender render. However, the self occlusion shading is reproduced with excellent accuracy.

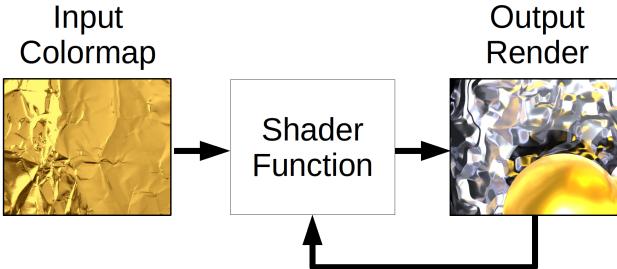


Figure 5: Blender material rendering process flow.

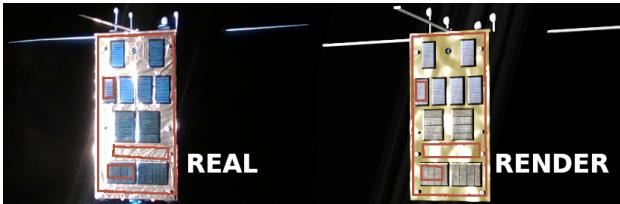


Figure 6: Micro-satellite real versus blender images.

Several regions of interest were compared in greater detail; the bounding boxes depicting these regions have also been drawn in Figure 6. As the images were input to the pose estimation pipeline in RGB format, it was suitable to analyze the RGB histograms. The histograms associated with Figure 6 are shown in the top row of Figure 7.

Even with representative material images as an input color map, the RGB histograms differed greatly. In the overall case, it can be seen that the real image contained a much greater percentage of maximum intensity pixels (value 1). In the MLI case, the blender render over-saturated many pixels. In the solar panel case, the real and render distributions were similarly located Gaussian distributions, although the real image distribution depicted a larger variance. Based on these observations, it would likely be a large challenge to generalize a pose estimation pipeline to real image inference when only trained on these render images.

Post Processing Image Augmentation—With the end objective being pose estimation, it was ideal to train the pose estimation pipeline to assess geometry. Building on previous work [13], a post processing image augmentation function was constructed to de-emphasize image textures to focus the network on target geometry. The function modified the images based on different physical and sensor-based sources of noise and was applied to both the real and render images:

1. Contrast Adjustment: To remove extraneous backdrop geometry, the render image intensities < 0.1 were set to 0.

2. Gaussian Blur: To guide the pose estimation pipeline to focus on larger geometric structures, a slight defocus was included. The MATLAB 2D Gaussian filtering function imgaussfilt() was used; the Gaussian distribution (σ) and filter size were scaled proportional to the target distance.

3. Artificial Bloom: Often, camera pixels are sequentially arrayed with anti-blooming drains at the end of a sequence. If a single pixel cell is saturated, it may overflow into adjacent cells [38]. 1D column Gaussian blurring was applied to intensities > 0.78 . Filter size and σ were scaled proportionally to the target distance.

Histogram	Overall	MLI	Solar Panel
Pre RGB	.32, .30, .25	.33, .37, .39	.45, .34, .30
Pre mean	0.291	0.359	0.362
Post RGB	.23, .17, .13	.11, .11, .23	.29, .25, .22
Post mean	0.176	0.148	0.254

Table 1: Battacharya Distance of Pre-Augmentation and Post-Augmentation Image Histograms (lower is better).

4. Random Artifact: To reduce the pose estimation pipeline reliance on specific image regions, a function was coded to add an arbitrary shaped polygon to the image. The function was passed 5 times over an image with a probability of 0.5 to add a polygon.

5. Saturated Bloom: Observable in Figure 6, high intensity locations were often accompanied by a gradient corona. This function replicated such behaviour.

6. Speckle Noise: To improve the generality of the pose estimation pipeline, this function globally introduced "speckle" noise to the image. The MATLAB function imnoise() was used here.

7. Sensor Saturation: Again, to improve the generality of the pose estimation pipeline, this function globally increased the image intensity by a random value in the range $[0, 0.3]$.

The corresponding augmented images are shown in Figure 8. Although not perfect, the augmented images include many additional sources of noise which forces a neural network to learn larger geometric features as opposed to more local material features. This process was used to great effect in [13]; the author was able to train a pose estimation pipeline on a similarly rendered dataset for performing inferences on real spacecraft images.

Augmented Render to Real Comparison—The Figure 8 augmented image RGB histograms can be seen in the bottom row of Figure 7. The improvement in real-to-render RGB histogram similarity is easily observable. To quantify the histogram similarity, the Battacharya distance metric [14] is introduced here:

$$d(H_1, H_2) = \sqrt{1 - \frac{1}{\sqrt{H_1 H_2 N^2}} \sum_{i=1}^N \sqrt{H_1(i) H_2(i)}} \quad (1)$$

Where H_1, H_2 are two histograms, $\overline{H_1}, \overline{H_2}$ are the mean of the two histograms, and N is the total number of histogram bins. The Battacharya distances for the pre-augmentation and post-augmentation image comparisons are summarized in Table 1. It is easily observable that the real and render image similarity was greatly improved.

H2A Second Stage

With a suitable render pipeline developed and assessed, the focus shifted to the H2A 2nd stage rocket body dataset.

The target orbital position was discretized from the Two Line Element (TLE) set: 33500, an H2A 2nd stage in approximately sun-synchronous orbit. The inclination and period of the orbit were 98.1° and 96.7 minutes, respectively. Three different sunlight vectors were modelled to approximate potential seasonal variations. The resulting orbit can be viewed

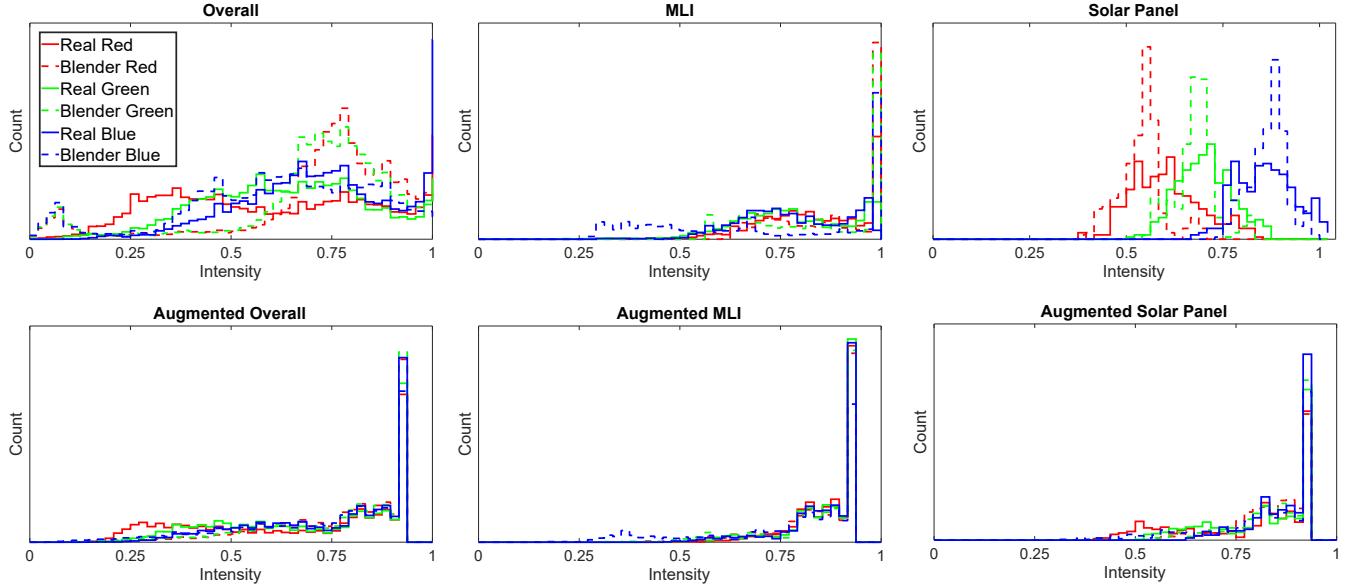


Figure 7: Micro-satellite real versus render histograms for three regions.
Top row: image histograms before post processing augmentation.
Bottom row: image histograms after post processing augmentation.



Figure 8: Augmented real versus blender images.

in Figure 9. In a target-local Hill frame, two tumble motions were simulated using a 4th order Runge-Kutta integrator with time steps of 0.1s: a **flatspin** and a **conic** motion with fixed nutation angle. An inertial matrix of $\begin{bmatrix} 4.1 & 0 & 0 \\ 0 & 4.1 & 0 \\ 0 & 0 & 1 \end{bmatrix} m$ was used. Up to three complete orbits were simulated for each case.

16,845 images were rendered across the two tumble motions, three sun vectors and a variety of motion time steps. An example of the flatspin case is shown in Figure 10 and the conic case is shown in Figure 11. Note that the sunlight intensity resulted in high contrast images with very little colour information.

4. POSE ESTIMATION NETWORK

The pose estimation architecture was aimed at simplicity rather than accuracy. A small and simple pose estimation pipeline enabled minimization and quantization studies as well as providing an excellent benchmark for future work implementing the pipeline on spacecraft hardware. This approach was inspired by the direct regression network design of [33]. The network architecture is depicted in Figure 12 and featured a ResNet backbone [39] with the final global pooling layer and activation functions removed to maintain

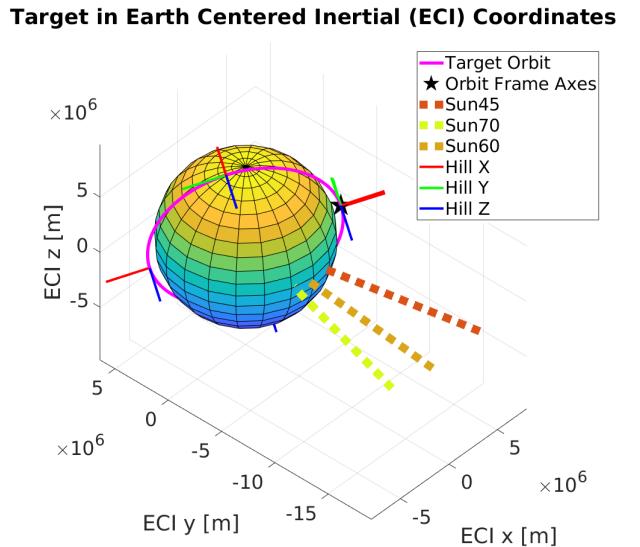


Figure 9: H2A 2nd stage simulated orbit.



Figure 10: Flatspin tumble case.



Figure 11: Conic tumble case.

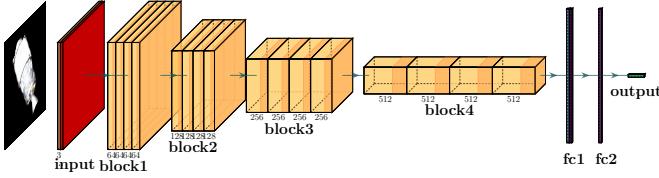


Figure 12: Overview of network architecture.

spatial resolution; the only remaining pooling occurred near the beginning of the network in the *input* block. A small fully connected network was post-fixed to regress the required output terms. However, our loss function and ground truth differed considerably from previous works.

Again, to maintain a simple post-processing averse design, the orientation vector was directly regressed without requiring PnP [31] [32] or probabilistic fitting [33] post processing. Instead an 8x1 ground truth vector was constructed containing image plane angles, target fixed frame angles and a signed 3 bit quadrant value indicating which quadrant the target is “pointed” towards; the details can be seen in Table 2 and Figure 13.

A weighted Mean Squared Error (MSE) loss function with control weights α , β and γ were used during training:

$$L(y, \hat{y}) = \frac{\alpha}{2} \sum_{i=1}^2 (y_i - \hat{y}_i)^2 + \frac{\beta}{3} \sum_{i=3}^5 (y_i - \hat{y}_i)^2 + \frac{\gamma}{3} \sum_{i=6}^8 (y_i - \hat{y}_i)^2 \quad (2)$$

Where y, \hat{y} are the the network output and ground truth and i is the vector encoding index. In this way, a higher weight was lent to the quadrant value at the start of training to help the network converge, while a higher weight was lent to the image plane values near the end of training to improve the accuracy.

While unorthodox, the image plane representation was a simple regression problem for a CNN whose hidden layers

Vector	Description	Reference
θ_{in}	Angle about z-axis	IP
θ_{out}	Angle about x-axis	IP
Θ	Pitch	OF
Ψ	Roll	OF
Ω	Yaw	OF
[111]	Value indicating quadrant	OF

Table 2: Ground Truth Vector Encoding.

IP - image plane
OF - object frame

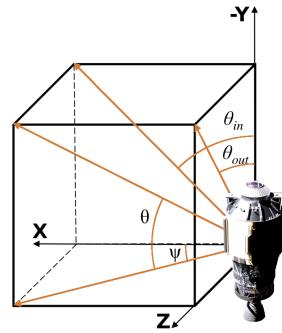


Figure 13: Output vector frames of reference.

Case	RMSE
Flatspin A In-plane Angle	16.22°
Flatspin B In-plane Angle	5.52°
Conic In-plane Angle	1.79°
Conic Out-plane Angle	1.04°

Table 3: Root Mean Square Error (RMSE) Results for Flatspin and Conic Motions

were inherently operating in the image plane domain. Again, the quadrant bit value was a relatively trivial regression problem for the neural network but helped greatly during initial training for convergence and also theoretically maintained utility for object tracking; these values could increase the stability of a hypothetical Kalman Filter tracking the object’s motion. Finally, the Euler angles, although a poor regression choice due to their order of operations, maintained utility as they could be directly calculated from the image plane representation as:

$$\theta_{in} = \arctan \left(\frac{\cos \Theta \cos \Psi}{\sin \Theta} \right) \quad (3)$$

$$\theta_{out} = \arctan \left(\frac{\cos \Theta \sin \Psi}{\sin \Theta} \right) \quad (4)$$

Pose Estimation Results

The pose estimation pipeline was trained for 25 epochs on images featuring two of the three sun vectors and then validated and tested on the third sun vector images. A closer inspection of the results can be found in the Section **Quantization Case Study**. A summary of the prediction performance for all cases is shown in Table 3. It is worth noting that the poorest network predictions occurred at the limits of the conic motion. Overall, the results were highly reliable and suitably accurate to perform motion predictions.

Limited Bandwidth Motion Estimation

Up till now, this work has described a dataset and corresponding pose predictions in ideal circumstances. Indeed, the contribution of a large part of the computer vision community is focused on algorithm improvement and optimization for well classified problems and datasets. However, as discussed previously, spacecraft pose estimation faces the unique challenge of bandwidth limitation.

Spacecraft attitude and motion estimation is a mature subject of study [40] [41] with still recent developments in the

Strategy	Flatspin A Est. Rate	Flatspin B Est. Rate
Ground Truth	1.0000 deg/s	1.0000 deg/s
(Ideal) All	1.0004 deg/s	0.9995 deg/s
(A) First/Last	1.0004 deg/s	1.0039 deg/s
(B) Last 36	0.9970 deg/s	0.9880 deg/s
(C) Every 120s	1.0004 deg/s	1.0001 deg/s

Table 4: Flatspin Rate Estimation Results for Image Sample Strategies

context of computer vision [42] [43]. However, the vast majority of these studies explicitly or implicitly assume a constant sample rate. Depending on the computer vision attitude determination mission specifics, maintaining a constant image sample rate may be neither possible or efficient.

During a hypothetical debris capture mission, the target motion must be accurately resolved both spatially and temporally. Consider the simple flatspin case depicted in Figure 10 with a rate of 1 deg/s. This motion was simulated for 1 orbit. Four simple image sample strategies were used: (Ideal) All images; (A) First and last image; (B) Last 36 images; and (C) An image every 120s. The rate was then simply estimated from a least-squares regression fit from the sampled images. The results are shared in Table 4. In the event that the ideal case is not possible, sampling over a longer period of time improved motion estimation accuracy. Sampling at the maximum possible sample rate for only the last 36 images consistently yielded the worst rate estimations.

Conversely, a high sample rate is required to exclude higher order motion harmonics. If an image is captured every 1 s, [1, 2 ... 10 ...] deg/s rates are all possible rate estimations. In conventional signal processing theory, a sampling rate sufficient to resolve the signal is referred to as the Nyquist sample rate [44] and equates to at least twice the observed harmonic motion. However, a non uniform sampling rate could theoretically exclude higher order harmonics as well.

From this discussion it can be determined that: sampling over a longer period of time will result in a higher estimator accuracy; and the images must be sampled with sufficient frequency to resolve the motion. Thus bandwidth or memory restricted space missions should likely tend towards non-uniform sampling motion estimators such as a non-uniform Kalman Filter [45].

5. QUANTIZATION CASE STUDY

Previously, bandwidth limited terrestrial-based post processing limitations have been discussed. However, the ideal case is to perform pose estimation locally; i.e., edge computing. One of the current major challenges is the large disconnect between pose estimation pipeline size and the available memory of typical spacecraft hardware. Current state-of-the-art pose estimation systems are orders of magnitude too large; Wide Depth Range has 51.6 million parameters [34], ZebraPose has 30.5 million parameters [18] and SO-POSE has 51.1 million parameters [46]. The implementation of these networks onto spacecraft hardware is not feasible without modification.

There are a number of methods to reduce network size. Pruning removes filters and weights of least importance from the network [47] [48]. Knowledge distillation uses a large

Case	32 bit	4 bit	2 bit
Testing Loss	435.1	567.1	1047.6
θ_{in} Error	1.89°	1.83°	3.04°
θ_{out} Error	0.88°	0.93°	1.99°

Table 5: Quantized Network Prediction: Single Image

network or ensemble of networks to train a smaller specific application network [49]. A particularly promising method is quantization, which reduces the bit representation of values in the network (Eg. reducing the 32 bit floating point values to 4 bit or 2 bit representations). Recently, quantization methods have even been applied to a spacecraft pose estimation network [23]. The Incremental Network Quantization (INQ) approach was adopted here [22] based on the implementation methodology of [50]. For convenience, our slightly modified quantization algorithm is summarized in Algorithm 1.

Algorithm 1 Incremental Network Quantization

```

1: Input: Training data
2: Input:  $\sigma_1, \sigma_2, \dots, \sigma_N$ 
3: Input:  $W_l : 1 \leq l \leq L$ 
4: for  $n = 1, 2, \dots, N_{\text{epochs}}$  do
5:   Update learning rate and policy
6:   for  $l = 1, 2, \dots, L_{\text{layers}}$  do
7:     From  $\sigma_n$  update  $A_l, A_l^q, T_l$ 
8:     Quantize the weights in  $A_l^q$ 
9:     Calculate feed-forward loss
10:    Update  $A_l$  values with gradient descent
11:   end for
12: end for
13: Output:  $W_l^q : 1 \leq l \leq L$ 

```

- σ_n is the fraction of weights to be quantized at epoch n
- W_l is the full precision model (untrained)
- W_l^q is the quantized model (trained)
- N is the number of epochs to train
- L is the number of layers in the network
- A_l is the unquantized weights at layer l
- A_l^q is the quantized weights at layer l
- T_l is a binary matrix specifying quantized weights, where $[W_l][T_l(0)] = A_l$ and $[W_l][T_l(1)] = A_l^q$

On the more challenging conic motion dataset, three networks were trained: a full precision 32 bit network, a 4 bit network and a 2 bit network. The pose estimation pipeline was trained for 750 epochs with a negative pseudo-logarithmic quantization schedule (e.g. 40% of the network was quantized in the first 100 epochs). Prediction results for the first 200 images are shared in Figure 14 with the corresponding prediction errors in Figure 15; the testing loss function (Equation 2) and angle error results are shared in Table 5.

The 2 bit network quantized all weights and filters to the set $[-1, 0, 1]$; there was a notable reduction in accuracy, doubling the testing loss and angle estimation errors. Observing figure 14, it can be seen that the worst performance occurred at the limits of the conic motion; thus a good follow up would be to explore more dynamic and extreme motion profiles. However, it is clear that the 2 bit network was still able to track the motion well. The 4 bit network exhibited performance extremely similar to the full 32 bit network, even demonstrating a slightly higher accuracy for the θ_{in} parameter.

Normally, a Kalman filter variant would be the tool of choice

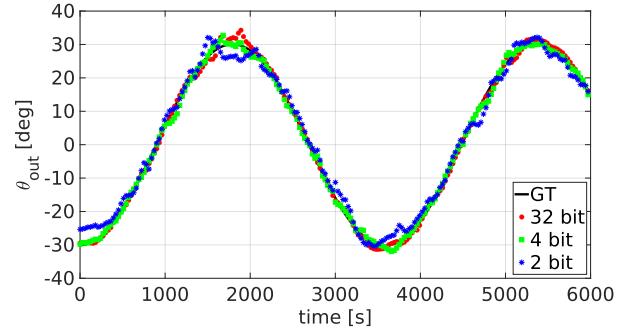
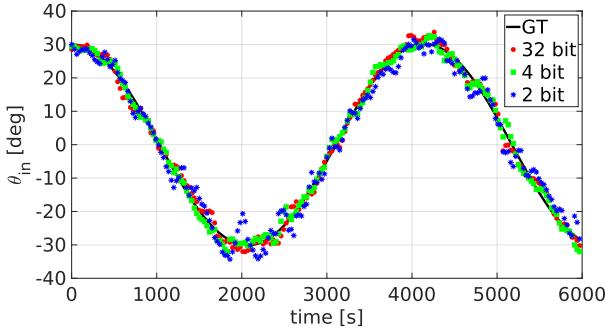


Figure 14: First 200 Network Predictions for the Conic Motion θ_{in} (left) and θ_{out} (right) Angles

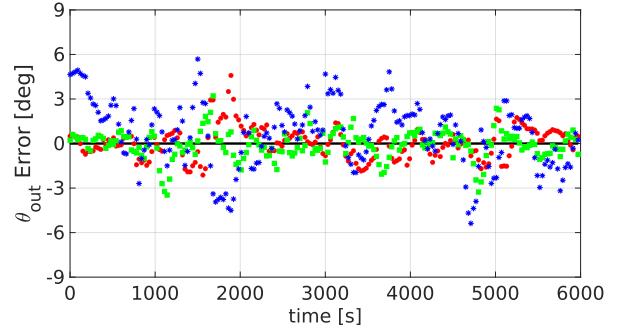
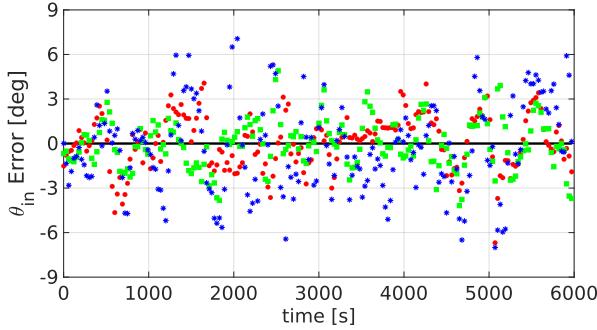


Figure 15: First 200 Network Prediction Errors for the Conic Motion θ_{in} (left) and θ_{out} (right) Angles

Fit Value	Ground Truth (GT)	32 bit	4 bit	2 bit
a [$^{\circ}$]	30.5	30.6	30.3	30.2
$1/b$ [s]	660.6	659.2	660.1	659.6
R^2 ($\uparrow +$)	-	0.99	0.99	0.98
RMSE ($\downarrow +$)	-	1.8	1.8	3.0

Table 6: Quantized Network Prediction: θ_{in} Motion Fit

Fit Value	Ground Truth (GT)	32 bit	4 bit	2 bit
a [$^{\circ}$]	30.6	31.0	30.3	29.5
$1/b$ [s]	571.4	571.1	571.1	572.1
R^2 ($\uparrow +$)	-	1.00	1.00	0.99
RMSE ($\downarrow +$)	-	0.8	0.9	1.8

Table 7: Quantized Network Prediction: θ_{out} Motion Fit

for motion estimation. However, as the conic motion was sinusoidal, a simple least squares regression to a sinusoidal curve fit was performed here. The three network predictions and ground truth data were fit to a four parameter sinusoid defined here in Equation 5.

$$\theta_{in/out}(t) = a * \sin(bt + c) + d \quad (5)$$

Where t is the time associated with that particular tumble simulation render, a is the magnitude term, b is the frequency term and c, d are offset terms. θ_{in} and θ_{out} fit results are shared in Tables 6 and 7. It can be easily observed that the 4 bit network motion prediction accuracy matched the 32 bit motion prediction accuracy. Additionally, depending on the mission constraints, the 2 bit network motion prediction results may also be sufficient. Clearly, the 4 bit network would be an excellent contender to be installed on spacecraft hardware. This network architecture had approximately 11 million parameters. Reducing the network from 32 bit to 4 bit (or 2 bit) resulted in a theoretical network size reduction of 44 MB to 5.5 MB (or 2.75 MB).

This work was performed in the context of the JAXA CRD2 mission. From publicly available information, a target model was constructed, an orbit was discretized and two tumble motions were simulated. From the simulated information we developed and share the new dataset. Although the dataset does not match the fidelity of state-of-the-art SPEED+, the dataset is directly applicable to an active debris capture mission. It is expected that research interest and developments surrounding the JAXA CRD2 mission will continue to grow and thus our dataset will continue to be of value. A render-to-real validation study was performed on the dataset and an image-augmentation post-processing pipeline (established from sensor and physics-based noise) was implemented, improving the render-to-real image comparison.

A small pose estimation network was constructed, optimized for small size and simplicity over accuracy; for future works, this network could be more feasibly installed on spacecraft hardware. The network was trained on data containing two sun-angle vectors and tested on a third sun-angle vector data. A simple case study explored bandwidth limited cases and concluded that non-uniform sampling motion estimators should be utilized.

A quantization study was performed comparing a full 32 bit precision to 4 bit and 2 bit networks. The quantized networks demonstrated sufficient accuracy for both single pose prediction and motion estimation tasks. The 4 bit network theoretically represented a factor of 8x in memory savings. The resulting 5.5 MB network could more easily be implemented on spacecraft hardware.

Follow up work will be focused on implementing the network on hardware and incorporating a non-uniform sampling motion estimator. In parallel, network performance across the domain gap will be analyzed using the micro-satellite physical and render models.

REFERENCES

- [1] Kent Tobiska, Marcin Pilinksi, Shaylah Mutschler, Kaiya Wahl, Jean Yoshii, Dave Bouwer, Piyush Mehta and Richard Licata. “Understanding variability in HASDM to support space traffic management”. In *Proceedings of the 23rd Advanced Maui Optical and Space Surveillance Technologies (AMOS) Conference*. AMOS, 2022.
- [2] Donald J. Kessler and Burton G. Cour-Palais. “Collision frequency of artificial satellites: the creation of a debris belt”. In *Journal of Geophysical Research*, 83(A6): pp. 2637–1646. 1978.
- [3] European Space Agency. “The current state of space debris”. *ESA Technical Report*. https://www.esa.int/Safety_Security/Space_Debris/The_current_state_of_space_debris. Accessed 11 February, 2021.
- [4] Japanese Space Agency. “Global efforts to deal with the problem of space debris”. *JAXA web article*. <https://global.jaxa.jp/article/2017/special/debris/nishida.html>. Accessed 11 February, 2021.
- [5] Orbital Debris Program Office. “LEGEND : 3D/OD evolutionary model”. *NASA ODPO web article*. <https://orbitaldebris.jsc.nasa.gov/modeling/legend.html>. Accessed 24 September, 2022.
- [6] NASA Orbital Debris Program Office. “Cosmos 1408 Break-up”. In *Orbital Debris Quarterly News*. 6(1): pp. 1–5. March, 2022.
- [7] Orbital Debris Research and Development Interagency Working Group Chair: Samuel Jones. “National orbital debris research and development plan”. *National Science and Technology Council Technical Report*. January, 2021.
- [8] Abid Murtaza, Syed Jahanzeb Hussain Pirzada, Tongge Xu and Liu Jian Wei. “Orbital debris threat for space sustainability and way forward”. *IEEE Access Multidisciplinary Open Access Journal*. 8: pp. 61000–61019. April, 2020.
- [9] K. Wormnes, R. Le Letty, L. Summerer, R. Schonengborg, O. Dubois-Matra, E. Luraschi, A. Cropp, H. Krag, J. Delaval. “ESA technologies for space debris remediation”. In *Proceedings of the 6th European Conference on Space Debris*. ESA, 2013.
- [10] Benjamin Bastidi Virgili and Holger Krag. “Analyzing the criteria for a stable environment”. In *Proceedings of the 2011 AAS/AIAA Astrodynamics Specialist Conference*. Advances in the Astronautical Sciences 142. August, 2011.
- [11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei. “Imagenet: a large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 248–255. 2009.
- [12] Mykhaylo Andriluka, Leonid Pischchulin, Peter Gehler and Bernt Schiele. “2D human pose estimation: new benchmark and state of the art analysis”. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June, 2014.
- [13] Andrew Price and Kazuya Yoshida. “A Monocular Pose Estimation Case Study: The Hayabusa2 Minerva-II2 Deployment”. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2021 AI for Space Workshop*. June, 2021.
- [14] Bi, Sifeng and Broggi, Matteo and Beer, Michael. “The role of the Bhattacharyya distance in stochastic model updating”. *Mechanical Systems and Signal Processing*. 117: pp. 437–452. 2019.
- [15] Mate Kisantai, Sumant Sharma, Tae Ha Park, Dario Izzo, Marcus Märkens and Simone D’Amico. “Satellite pose estimation challenge: dataset, competition design and results”. *IEEE Transactions on Aerospace and Electronic Systems Journal*. 56(5): pp. 4083–4098. April, 2020.
- [16] Jonathan Dupuy and Wenzel Jakob. “An adaptive parameterization for efficient material acquisition and rendering”. In *Proceedings of ACM Transactions on Graphics*. 37(6): pp. 1–14. December, 2018.
- [17] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan and Dieter Fox. “PoseCNN: a convolutional neural network for 6D object pose estimation in cluttered scenes”. In *Proceedings of the Robotics: Science and Systems Conference 2018*. June, 2018.
- [18] Yongzhi Su, Mahdi Saleh, Torben Fetzer and Jason Rambach. “ZebraPose: coarse to fine surface encoding for 6DoF object pose estimation”. In *Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. March, 2022.
- [19] Jason L. Forshaw, Guglielmo S. Aglietti, Nimal Navarathinam, Haval Kadhem, Thierry Salmon, Aurélien Pisseloup, Eric Joffre, Thomas Chabot, Ingo Retat, Robert Axthelm, Simon Barraclough, Andrew Ratcliffe, Cesar Bernal, François Chaumette, Alexandre Pollini and Willem H. Steyn. “RemoveDEBRIS: An in-orbit active debris removal demonstration mission”. In *Acta Astronautica* 127: pp 448–463. October–November, 2016.
- [20] Robin Biesbroek, Sarmad Aziz, Andrew Wolahan, Stefano Cipolla, Muriel Richard-noça and Luc Piguet. “The ClearSpace-1 mission: ESA and ClearSpace team up to remove debris”. In *Proceedings of the 8th European Conference on Space Debris*. 2021.
- [21] JAXA. “Commercial removal of debris demonstration (CRD2)”. *Web article*. <https://www.kenkai.jaxa.jp/eng/research/crd2/crd2.html>. Accessed 27 September, 2022.
- [22] Aojun Zhou, Anbang Yao, Yiwen Guo, Lin Xu and Yurong Chen. “Incremental network quantization: towards lossless CNNs with low-precision weights”. In *Proceedings of the 2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. February, 2017.
- [23] Shunli Wang, Shuaibing Wang, Bo Jiao, Dingkang Yang, Liuzhen Su, Peng Zhai, Chixiao Chen and Li-hua Zhang. “CA-SpaceNet: counterfactual analysis for 6D pose estimation in space”. In *Proceedings of the*

- [24] Simon Nolet. “The SPHERES Navigation System: from Early Development to On-Orbit Testing. In *AIAA Guidance, Navigation and Control Conference and Exhibition*. 2007. 8,9,10
- [25] Swati Mohan, Alvar Saenz-Otero, Simon Nolet, David W. Miller and Steven Sell. “SPHERES Flight Operations Testing and Execution.” *Acta Astronautica*. 2009.
- [26] Brent E. Tweddle, Elias Müggler, Alvar Saenz-Otero and David W. Miller. “The Spheres Vertifo Goggles: Vision Based Mapping and Localization Onboard the International Space Station”. In *Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation in Space*. 2012.
- [27] Michael C. O’Connor, Brent E Tweddle, Jacob G. Katz, Konrad Makowka, Alvar Saenz-Otero and David W. Miller. “Visual-Inertial Estimation and Control for Inspection of a Tumbling Spacecraft: Experimental Results from the International Space Station.” In *AIAA Guidance, Navigation and Control Conference*. 2012.
- [28] Haopeng Zhang, Zhiguo Jiang, Yuan Yao and Gang Meng. “Vision-Based Pose Estimation for Space Objects by Gaussian Process Regression”. In *Proceedings of IEEE Aerospace Conference*. 2015.
- [29] ESA Advanced Concept Team and Stanford SLAB. “Kelvins pose estimation challenge”. *Kelvins - ESA’s Advanced Concepts Competition Website*. <https://kelvins.esa.int/satellite-pose-estimation-challenge/home/>. Accessed 18 February, 2021.
- [30] Sumant Sharma. “Pose estimation of uncooperative spacecraft using monocular vision and deep learning”. *Ph.D. Thesis*, Standford University, Stanford, California, USA. August, 2019.
- [31] Bo Chenm Jiewei Cao, Álvaro Parra and Tat-Jun Chin. “Satellite pose estimation with deep landmark regression and non-linear pose refinement”. In *Proceedings of 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. October, 2019.
- [32] Tae Ha Park, Sumant Sharma and Simone D’Amico. “Towards robust learning-based pose estimation of non-cooperative spacecraft”. In *Proceedings of the AAS/AIAA Astrodynamics Specialist Conference*. August, 2019.
- [33] Pedro F. Proençá and Yang Gao. “Deep learning for spacecraft pose estimation from photorealistic rendering”. In *arxiv arXiv:1907.04298v2*. August, 2019.
- [34] Yinlin Hu, Sébastien Speierer, Wenzel Jakob, Pascal Fua and Mathieu Salzmann. “Wide-depth-range 6D object pose estimation in space”. In *Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June, 2021.
- [35] Tae Ha Park, Marcus Märtens, Gurvan Lecuyer, Dario Izzo and Simone D’Amico. “SPEED+: next-generation dataset for spacecraft pose estimation across domain gap”. In *Proceedings of the 2022 IEEE Aerospace Conference (AERO)*. March, 2022.
- [36] Japan Aerospace Exploration Agency. “About H-IIA launch vehicle”. *Web Article*. <https://global.jaxa.jp/projects/rockets/h2a/>. Accessed 1 November, 2021.
- [37] Rebecca Lindsey. “Climate and Earth’s energy budget”. *Web article*. NASA Earth Observatory. <https://earthobservatory.nasa.gov/features/EnergyBalance>. January, 2009.
- [38] Thomas J. Fellers and Michael W. Davidson. “Concepts in digital imaging technology: CCD saturation and blooming”. *Web Article*. Florida State University: National High Magnetic Field Laboratory 1800 East Paul Dirac Dr., Tallahassee, Florida, USA, 32310. Accessed 31 August, 2020.
- [39] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. “Deep residual learning for image recognition”. In *Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June, 2016.
- [40] E.J. Lefferts, F.L. Markley and M.D. Shuster. “Kalman filtering for spacecraft attitude estimation”. In *Proceedings of AIAA 20th Aerospace Sciences Meeting*. Paper 82-0070. 21 May, 1982.
- [41] Itzhack Y. Bar-Itzhack and Julie Deutschmann. “Comparison and testing of extended Kalman filters for attitude estimation of the Earth radiation budget satellite”. In *Proceedings of the AIAA/AAS Astrodynamics Conference*. 1 January, 1990.
- [42] János Takátsy, Tamás Bozóki, Gergely Dálya, Kornél Kapás, László Mészáros and András Pál. “Attitude determination for nano-satellites - II. Dead reckoning with a multiplicative extended Kalman filter”. In *Exp Astron* 53: pp. 209–223. 2022.
- [43] Farhad Aghili. “Robust 3D vision for autonomous robots”. *Arxiv* <https://arxiv.org/abs/2208.12925>. 30 August 2022.
- [44] Steven W. Smith. “The scientist and engineer’s guide to digital signal processing”. *book ISBN 0-9660176-7-6*. California Technical Publishing. 1999. **Or other similar signal processing theory sources**.
- [45] Weihua Li, Sirish L. Shah and Deyun Xiao. “Kalman filters for non-uniformly sampled multirate systems”. In *IFAC Proceedings Volumes*. 38(1): pp. 99–104. 2005.
- [46] Yan Di, Fabian Manhardt, Gu Wang and Xiangyang Ji. “SO-Pose: exploiting self-occlusion for direct 6D pose estimation”. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. October, 2021.
- [47] Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila and Jan Kautz. “Pruning convolutional neural networks for resource efficient inference”. In *Proceedings of the International Conference on Learning Representations (ICLR)*. 2017.
- [48] David Blalock, Jose Javier Gonzalez Ortiz, Jonathan Frankle and John Guttag. “What is the state of neural network pruning?”. In *Proceedings of the Matching Learning and Systems*. 2020.
- [49] Geoffrey Hinton, Oriol Vinyals and Jeff Dean. “Distilling the knowledge in a neural network”. In *Proceedings of the Neural Information Processing Systems Deep Learning Workshop*. 2014.
- [50] Saqib Javed. “Hardware-friendly mixed-precision neural networks”. *Masters Thesis*. Department of Information Technology and Electrical Engineering Swiss Federal Institute of Technology Zurich.

BIOGRAPHY



Andrew Price received his M.A.Sc degree from Carleton University, Canada and is currently a Ph.D. candidate at Tohoku University, Japan. He worked for five years as a research officer at the Canadian National Research Council performing qualification testing on satellites and aircraft. His research interests include computer vision, machine learning and spacecraft robotics.



Kentaro Uno received his B.Eng., M.E., and Ph.D. degrees in Engineering from Tohoku University, Japan. He worked as a software engineer intern in the Google Lunar XPRIZE participant: HAKUTO/ispace inc. and has studied in ETH Zurich as a research intern, Switzerland. Since 2021, he works as an Assistant Professor in the Department of Aerospace Engineering, Tohoku University. His research interests include mobile service robotics in extreme environments such as space and unstructured terrain.



Swapnil Parekh obtained his Master's degree from International Space University, France, and B.Eng from Institute of Infrastructure Technology Research and Management (ITRAM), India. His master's project focused on the feasibility analysis of the ChipSat for Mars atmosphere characterization. His research interests include space robotics, micro satellites, and electric propulsion.



Til Reichelt studies mechanical engineering, specializing in aerospace engineering, at TU Dresden, Germany. While taking part in an exchange program with Tohoku University, he got the opportunity to contribute to this paper. His research interests include space propulsion and exploration.



Kazuya Yoshida received his doctorate from Tokyo Institute of Technology in 1990. He is currently a professor at Tohoku University, leading the Space Robotics Laboratory, but also teaches for the International Space University Master of Space Studies program in Strasbourg, France. His technical contributions can be found in the ETS-VII, Hayabusa, and RISING/RISING-2 spacecraft as well as the Fukushima power plant incident robot: QUINCE. He has been a member of IEEE since 1990.