AAS 19-350

POSE ESTIMATION FOR NON-COOPERATIVE SPACECRAFT RENDEZVOUS USING NEURAL NETWORKS

Sumant Sharma[*], **Simone D'Amico**[†]

This work introduces the Spacecraft Pose Network (SPN) for on-board estimation of the pose, i.e., the relative position and attitude, of a known non-cooperative spacecraft using monocular vision. In contrast to other state-of-the-art pose estimation approaches for spaceborne applications, the SPN method does not require the formulation of hand-engineered features and only requires a single grayscale image to determine the pose of the spacecraft relative to the camera. The SPN method uses a Convolutional Neural Network (CNN) with three branches to solve the problems of relative attitude and relative position estimation. The first branch of the CNN bootstraps a state-of-the-art object detection algorithm to detect a 2D bounding box around the target spacecraft in the input image. The region inside the 2D bounding box is then used by the other two branches of the CNN to determine the relative attitude by initially classifying the input region into discrete coarse attitude labels before regressing to a finer estimate. The SPN method then uses a novel Gauss-Newton algorithm to estimate the relative position by using the constraints imposed by the detected 2D bounding box and the estimated relative attitude. The secondary contribution of this work is the generation of the Spacecraft PosE Estimation Dataset (SPEED), which is used to train and evaluate the performance of the SPN method. SPEED consists of synthetic as well as actual camera images of a mock-up of the Tango spacecraft from the PRISMA mission. The synthetic images are created by fusing OpenGL-based renderings of the spacecraft's 3D model with actual images of the Earth captured by the Himawari-8 meteorological satellite. The actual camera images are created using a 7 degrees-of-freedom robotic arm, which positions and orients a vision-based sensor with respect to a full-scale mock-up of the Tango spacecraft. Custom illumination devices simulate the Earth albedo and Sun light with high fidelity to emulate the illumination conditions present in space. The SPN method, trained only on synthetic images, produces degree-level relative attitude error and cm-level relative position errors when evaluated on the actual camera images not used during training.

INTRODUCTION

The on-board estimation of the pose, i.e., the relative position and attitude, of a known non-cooperative spacecraft using a monocular camera is a key-enabling technology for current and future on-orbiting servicing and debris removal missions such as the RemoveDEBRIS mission by Surrey Space Centre,[1] the Phoenix program by DARPA,[2] and the Restore-L mission by NASA.[3] The knowledge of the current pose of the target spacecraft during the proximity operations of these missions allows real-time approach trajectory guidance and control. In contrast to systems based on LiDAR and stereo camera sensors, monocular navigation ensures pose estimation under low mass and power requirements, making it a natural sensor candidate for future formation-flying missions especially employing small satellites.

[*] Ph.D. Candidate, Department of Aeronautics & Astronautics, Stanford University, Stanford, California, USA.

[†] Assistant Professor, Department of Aeronautics & Astronautics, Stanford University, Stanford, California, USA.

Prior demonstrations of close-range pose estimation have utilized image processing based on hand-engineered features[4–12] and an a-priori knowledge of the pose.[13–16] A key strength for many of these methods is their use of the perspective transformation between the scene and the image to hypothesize and test feature correspondences detected in the 2D image and the known 3D model of target spacecraft. However, the formulation of specific features is not scalable to spacecraft of different structural and physical properties as well as not robust to the dynamic illumination conditions of space. Secondly, the a-priori knowledge of the pose of the target is not always available due to mission operations constraints nor desirable when full autonomy is required.

Recent advancements in pose estimation techniques for terrestrial applications have relied on deep learning algorithms. Broadly, these algorithms bypass the classical image processing based pipeline and instead attempt to learn the non-linear transformation between the two-dimensional input image space and the six-dimensional output pose space in an end-to-end fashion. The deep learning-based methods either discretize the pose space and solve the resulting classification problem[17–20] or directly regress the relative pose from the input image.[21–23] Render-for-CNN[18] demonstrated the use of rendered images to train a convolutional neural network for viewpoint estimation on actual camera images. The viewpoint estimation problem is casted as classifying the camera rotation parameters into fine-grained bins. PoseCNN[23] used separate branches of a convolutional neural network to predict semantic labels, object centers in the 2D image, and object rotation using direct regression. However, the approach is not accurate enough without further refinement using the iterative closest point method.[24] Classification-based approaches rely on the fine discretization of the pose space into a large number of pose labels in order to achieve reasonable pose accuracy. On the other hand, direct regression-based approaches require careful choice of parameters to avoid unpredictable behavior while learning the transformation between the input two-dimensional pixel information and the output regression parameters describing the six-dimensional pose space. In addition, the applicability of these algorithms to space imagery is not trivial due to two reasons. Firstly, unlike terrestrial applications, spaceborne navigation cameras are challenged with quickly varying illumination conditions and capture imagery with low signal-to-noise ratio and high contrast. Secondly, the massive datasets necessary to train machine learning algorithms are typically not available for spaceborne navigation.

The primary contribution of this work is the introduction of the Spacecraft Pose Network (SPN), a new deep learning-based pose estimation method. As shown in Figure 1, the SPN method uses a convolutional neural network to estimate the relative position and relative attitude in a decoupled fashion from a single grayscale image. One branch of the convolutional neural network is used to detect a 2D bounding box around the target spacecraft in the input image. The other two branches are used to estimate the relative attitude using a hybrid discrete-continuous method. The relative attitude and the 2D bounding box are then combined with geometrical constraints of the perspective transformation to estimate the relative position using the Gauss-Newton algorithm. In contrast to current deep learning-based techniques, the relative attitude accuracy provided by the SPN method is not limited by the level of discretization of the pose space and it explicitly uses the geometrical knowledge of the perspective transformation in the estimation of relative position. As compared with techniques comparing image features with features of a known target spacecraft model onboard, the SPN method provides a pose estimate from a single image without requiring a long initialization phase or favorable relative translation motion. Further, due to the decoupling of the relative attitude and position estimation and the use of transfer learning, the SPN method can be used to infer the pose of the target spacecraft through training on a relatively small number of synthetic

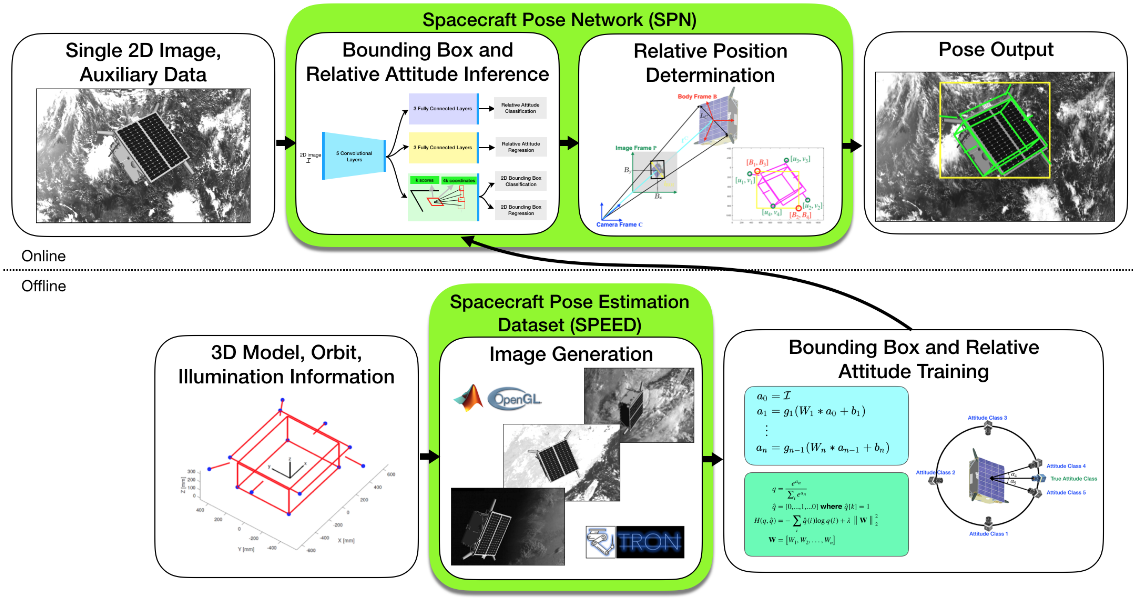
Single 2D Image, Auxiliary Data
Spacecraft Pose Network (SPN)
Bounding Box and Relative Attitude Inference
2D image $\mathcal{I}$
5 Convolutional Layers
3 Fully Connected Layers
Relative Attitude Classification
3 Fully Connected Layers
Relative Attitude Regression
k scores
4k coordinates
2D Bounding Box Classification
2D Bounding Box Regression
Relative Position Determination
Body Frame B
L_C
Image Frame P
B_y
B_x
Camera Frame C
$[B_1, B_3]$
$[u_3, v_3]$
$[u_1, v_1]$
$[u_2, v_2]$
$[u_4, v_4]$
$[B_2, B_4]$
Pose Output
Online
Offline
3D Model, Orbit, Illumination Information
Spacecraft Pose Estimation Dataset (SPEED)
Image Generation
OpenGL
TRON
Bounding Box and Relative Attitude Training
$a_0 = \mathcal{I}$
$a_1 = g_1(W_1 * a_0 + b_1)$
$\vdots$
$a_n = g_{n-1}(W_n * a_{n-1} + b_n)$
$q = \frac{e^{a_n}}{\sum_i e^{a_n}}$
$\hat{q} = [0,...,1,...0]$ where $\hat{q}[k] = 1$
$H(q,\hat{q}) = -\sum_i \hat{q}(i)\log q(i) + \lambda \parallel \mathbf{W} \parallel_2^2$
$\mathbf{W} = [W_1, W_2, \dots, W_n]$
Attitude Class 3
Attitude Class 2
Attitude Class 4
True Attitude Class
Attitude Class 5
Attitude Class 1
α_4
α_5

Figure 1. Modules of the proposed SPN method, which takes as input a 2D image and a trained convolutional neural network for relative attitude and position determination.

images of the same spacecraft.

The secondary contribution of this work is the creation of the Spacecraft Pose Estimation Dataset (SPEED), which consists of high-fidelity imagery involving close proximity operations around a tumbling spacecraft. The dataset contains 15,300 images from two sources: a purely software-based Augmented Reality (AR) source that fuses synthetic and actual space imagery, and a purely reality-based source that uses an actual camera sensor to capture images of a mock-up spacecraft under high-fidelity illumination conditions. The convolutional neural network of the SPN method is trained only on a portion of these AR images while it is tested on the remaining AR images as well as the actual camera images of SPEED. SPEED will also be made publicly available to the community to enable a fair comparison of the state-of-the-art pose estimation techniques through a competition on pose estimation for spaceborne applications organized in collaboration with the European Space Agency.[25]

The paper is organized as follows: Section 2 describes the framework for the SPN method; Section 3 describes the SPEED image generation; Section 4 describes the experiments conducted to validate the performance of SPN method; and Section 5 presents conclusions from this work as well as directions for further research and development.

SPACECRAFT POSE NETWORK

Formally, the problem statement for this work is the estimation of the attitude and position of the camera frame, C, with respect to the body frame of the target spacecraft, B. As shown in Figure 2, $\mathbf{t_{BC}}$ is the relative position of the origin of the target's body reference frame w.r.t. the origin of the camera's reference frame. Similarly, $\mathbf{q_{BC}}$ is the quaternion associated with the rotation that aligns the target's body reference frame with the camera's reference frame.

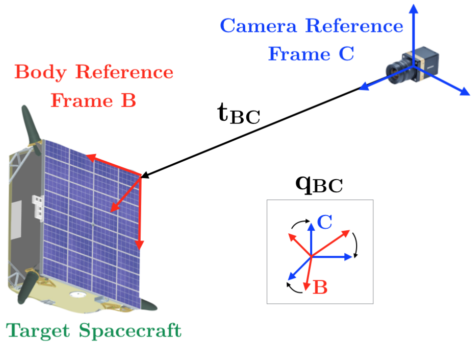

Figure 2. Definition of the reference frames, relative position, and relative attitude.

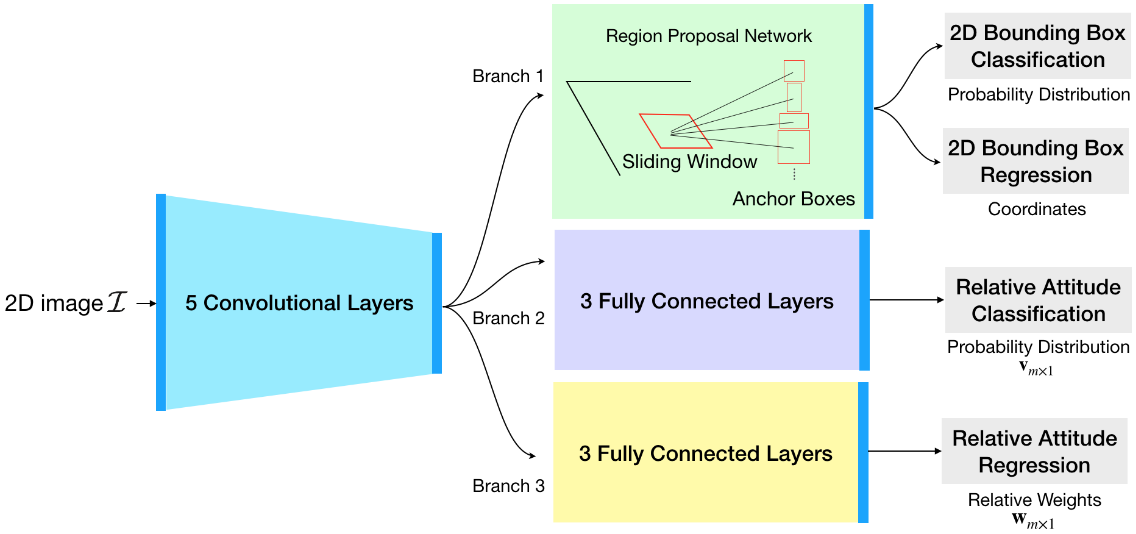

Figure 3. Illustration of the convolutional neural network used in the SPN method. Branch 1 uses the region proposal network[26] to detect a 2D bounding box around the target spacecraft while Branches 2 and 3 of the network are used in a hybrid classification-regression fashion to obtain the relative position.

As shown in Figure 3, the SPN method uses three separate branches of a convolutional neural network to estimate $\mathbf{t_{BC}}$ and $\mathbf{q_{BC}}$. Branch 1 is used to detect a 2D bounding box in the image around the target spacecraft. The output of the convolutional layers corresponding to the detected 2D bounding box is used as an input to Branches 2 and 3 to obtain an estimate of the relative attitude, $\tilde{\mathbf{q}}_{\mathbf{BC}}$. Finally, the 2D bounding box and $\tilde{\mathbf{q}}_{\mathbf{BC}}$ are input to the Gauss-Newton algorithm to estimate the relative position as $\tilde{\mathbf{t}}_{\mathbf{BC}}$. The estimation of the relative position and relative attitude are discussed in detail in the following subsections.

Relative Position Estimation

The relative position estimation begins with the detection of a 2D bounding box in the image around the target spacecraft using the region proposal network.[26] The detected 2D bounding box is not only important to effectively remove the background from the image before relative attitude estimation but makes the SPN method extensible to perform pose estimation for multiple objects or spacecraft components from the same image. The region proposal network used in the SPN method is an "off-the-shelf" object detection algorithm that takes the output of the five convolutional layers as an input. This network uses a sliding window approach on the output of the convolutional layers to produce region proposals based on predefined anchor boxes. Each anchor box is centered at the sliding window in question, and is associated with a scale and aspect ratio. Typically, 3 scales and 3 aspect ratios are used, yielding $k = 9$ anchor boxes at each sliding position. Therefore, for each sliding window location, the region proposal network outputs k probabilities (or scores) of whether the target spacecraft is present and regresses to $4k$ coordinates of the corresponding k 2D bounding boxes. Finally, the 2D bounding box associated with the highest probability is provided as the output. The reader can find further implementation details in the original paper.[26] Even though the SPN method uses the region proposal network in its current implementation, it can be easily swapped with another state-of-the-art object detection algorithm[27] based on specific computation runtime, storage, and accuracy requirements.

The resulting 2D bounding box estimated by the region proposal network and the relative attitude estimated by Branches 2 and 3 of the SPN method are combined with geometrical constraints to estimate the relative position using the Gauss-Newton algorithm. The SPN method uses the fact that the perspective projection of the 3D wireframe model of the target spacecraft must fit tightly within the detected 2D bounding box.

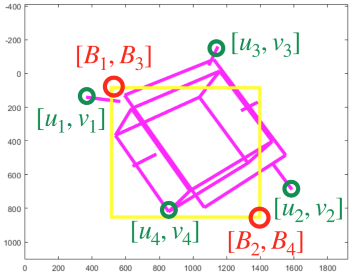

Figure 4. Schematic of the projection of the 3D wireframe model of the target (pink) and the estimated bounding box (yellow) in the image plane.

As shown in Figure 4, the 2D bounding box can be parametrized by its top-left coordinates, (B_1, B_3), and its bottom-right coordinates, (B_2, B_4), defined in the image plane. Given the estimated relative attitude parametrized as a rotation matrix, $R(\tilde{\mathbf{q}}_{\mathbf{BC}})$, the camera focal lengths, (f_x, f_y), and the camera principal points, (c_x, c_y), the 3D points defined in the body frame of the target spacecraft, $\mathbf{X}_i$, can be projected into the image plane using the perspective equation,

$$\begin{bmatrix} u_i w_i \\ v_i w_i \\ w_i \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R(\tilde{\mathbf{q}}_{\mathbf{BC}}) & \mathbf{t}_{\mathbf{BC}} \end{bmatrix} \mathbf{X}_i. \quad (1)$$

The constraint that the perspective projection of the 3D wireframe model fits tightly within the 2D bounding box requires that the four extremal projected points, (u_i, v_i), be as close as possible to the four edges of the 2D bounding box. Specifically, the four extremal projected points are the "left-most", "right-most", "top-most", and "bottom-most" points, respectively. For example, (u_1, v_1) is the "left-most" projected point and it is constrained to be as close as possible to B_1, the coordinate representing the left edge of the 2D bounding box. Mathematically, the SPN method solves the following minimization problem subject to the constraint posed by Equation 1

$$\underset{\mathbf{t_{BC}}}{\text{minimize}} \quad \sum_{i=1}^{2}(u_i(\mathbf{t_{BC}}) - B_i)^2 + \sum_{i=3}^{4}(v_i(\mathbf{t_{BC}}) - B_i)^2. \tag{2}$$

The minimization problem is solved using the Gauss-Newton algorithm, which requires an initial guess of $\mathbf{t_{BC}}$. The SPN method uses the detected 2D bounding box and the characteristic length of the 3D wireframe model to provide this initial guess.

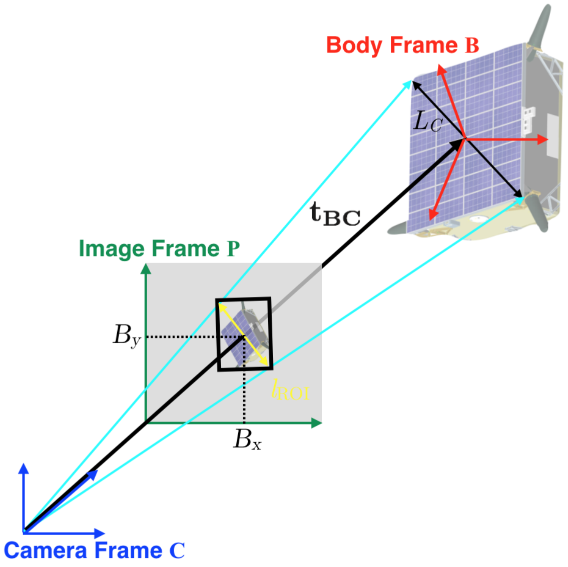

Figure 5. Calculation of the relative position using the 2D bounding box.

Figure 5 shows that the knowledge of the diagonal characteristic length, L_C, of the spacecraft 3D model and the diagonal length of the detected 2D bounding box, l_{ROI}, can be used to obtain $\mathbf{t_{BC}}$. In particular, a coarse estimate of the distance to the target spacecraft from the origin of the camera frame is

$$||\mathbf{t_{BC}}||_2 = \frac{\left(\frac{f_x+f_y}{2}\right) L_C}{l_{\text{ROI}}}. \tag{3}$$

Assuming that the origin of the target spacecraft body frame, B, lies on the ray projected from the origin of the camera frame, C, towards the center of the detected 2D bounding box, (B_x, B_y), azimuth and elevation angles, (α, β), can be derived using (c_x, c_y) and (B_x, B_y)

$$\alpha = \tan^{-1}\left(\frac{B_x - c_x}{f_x}\right) \tag{4}$$

$$\beta = \tan^{-1}\left(\frac{B_y - c_y}{f_y}\right). \tag{5}$$

Finally, the coarse relative position used as the initial guess in the Gauss-Newton algorithm is

$$\mathbf{t_{BC}} = \begin{bmatrix} \cos(\alpha) & 0 & -\sin(\alpha) \\ 0 & 1 & 0 \\ \sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\beta) & \sin(\beta) \\ 0 & -\sin(\beta) & \cos(\beta) \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ ||\mathbf{t_{BC}}||_2 \end{bmatrix}. \tag{6}$$

Relative Attitude Determination

The layers of a convolutional neural network transform the raw pixel information, $\mathcal{I}$, to gradually more abstract representations using predefined non-linear functions that contain unknown coefficients. The output of the network is then constrained to minimize a loss function that represents a discrepancy or error between the output of the network and the expected output for a set of training samples (supervised learning). For example, the network used in the SPN method can be represented by

$$\begin{aligned} a_0 &= \mathcal{I} \\ a_1 &= g_1(W_1 * a_0 + b_1) \\ &\vdots \\ a_n &= g_{n-1}(W_n * a_{n-1} + b_n) \end{aligned} \tag{7}$$

where W_i are the unknown weights for the i-th layer, b_i are the unknown biases for the i-th layer, and g_i are the known non-linear functions for the i-th layer. For relative attitude estimation, the network's output is expected to be values defined on a continuous non-Euclidean space, prohibiting the direct use of a typical L2 loss function. To handle this problem, other authors have proposed several loss functions[23,28,29] that directly regress to a relative attitude estimate, however, these have limited accuracy and require further refinement.[30] Instead, the SPN method uses two branches of fully-connected layers that share the output of five preceding convolutional layers as their inputs. In order to be robust to the background in the images, the features output by the final convolutional layer associated with the detected 2D bounding box are selected using the RoI pooling layer.[31] Using these features as input, Branch 2 performs a classification task to find the closest predefined attitude classes that describe the input image. Branch 3 uses these features as input to perform a regression task to find the relative weights of the attitude classes identified in Branch 2. For example, Figure 6 shows a simplified example where one dimension of the relative attitude has been discretized. For this case, Branch 2 is expected to find attitude classes 4 and 5 as the two adjacent classes that best describe the input image while Branch 3 is expected to find relative weights as a function of the angular differences α_4 and α_5 associated with the corresponding attitude classes. The sizes of the five convolutional layers and both sets of the three fully connected layers have been adopted from the Zeiler and Fergus model architecture.[32]

The attitude classes are parametrized as m unit quaternions representing uniformly distributed random rotations in the $SO(3)$ space. Algorithm 1 shows the subgroup algorithm[33] used to obtain these random rotations. The algorithm amounts to multiplying a uniformly distributed element from the subgroup of planar rotations with a uniformly distributed coset represented by the rotations pointing the z-axis in different directions.

To determine which of the m predefined attitude classes are the closest to the given image, Branch 2 is tasked to output an $m \times 1$ vector $\mathbf{v}$, which represents a probability distribution. Each entry $\mathbf{v}_j$ is the probability whether the attitude class in question is one of the n closest attitude classes to

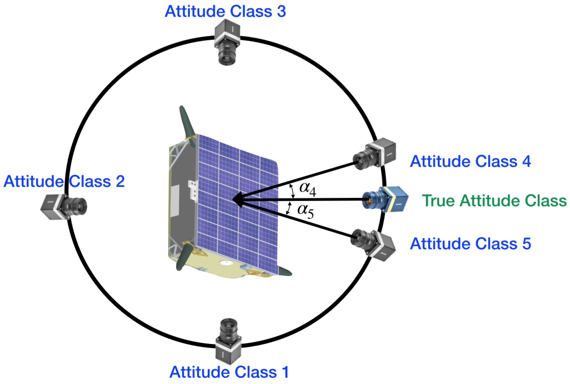

Figure 6. Visualization of the relative attitude discretization in a single dimension. Here attitude classes 4 and 5 are adjacent classes closest to the ground truth attitude.

Algorithm 1: Computes m uniformly distributed random rotations parametrized as unit quaternions.

```
Input  : Integer $m$
Output: Matrix quats
$x_0 \leftarrow \text{rand}(m)$ // get m uniformly distributed float values ∈ [0,1]
$x_1 \leftarrow \text{rand}(m)$
$x_2 \leftarrow \text{rand}(m)$
$\theta_1 = 2 * \pi * x1$
$\theta_2 = 2 * \pi * x2$
$s_1 = \sin(\theta_1)$
$s_2 = \sin(\theta_2)$
$c_1 = \cos(\theta_1)$
$c_2 = \cos(\theta_2)$
$r_1 = \sqrt{1 - x_0}$
$r_2 = \sqrt{x_0}$
quats = $[s_1 .* r_1, c_1 .* r_1, s_2 .* r_2, c_2 .* r_2]$
```

the relative attitude of the given image. Note that both m and n are hyper-parameters. Closeness is defined by the angular difference between the unit quaternion representing the attitude class, $\mathbf{q}_i$, and the unit quaternion representing the ground truth value of the relative attitude, $\mathbf{q}_{\mathbf{BC}}$. To ensure that $\mathbf{v}$ is a valid probability distribution, the output of the final layer of the Branch 2 is passed through the softmax function

$$\sigma(\mathbf{v})_j = \frac{e^{\mathbf{v}_j}}{\sum\limits_{k=1}^{m} e^{\mathbf{v}_j}} \quad \forall j \in [1, m] \tag{8}$$

Then the loss function for Branch 2, L_{class}, representing the difference between the branch output and the expected probability distribution, $\tilde{\mathbf{v}}$, can be written as

$$L_{\text{class}} = -\sum_{j=1}^{m} \tilde{\mathbf{v}}_j \log(\sigma(\mathbf{v})_j) + \lambda ||\mathbf{W_1}||_2^2, \tag{9}$$

where $\mathbf{W_1}$ represents the weights of the final three layers of Branch 2 and λ is a scalar representing the strength of the L2 regularization. The aim of the L2 regularization is to penalize the existence of large weights and prevent overfitting to the training examples. Note that the entries of $\tilde{\mathbf{v}}$ corresponding to the n closest attitude classes is set to $1/n$ while the rest of the entries are set to zero.

To estimate the relative attitude for the given image, Branch 3 is tasked to output an $m \times 1$ vector, $\mathbf{w}$, containing weights for the n closest attitude classes identified in Branch 2. The weights for the remaining $m - n$ attitude classes are output but not used during either training or inference. The weights of the n closest attitude classes are passed through the softmax function such that their sum adds to unity. Let Ω be the vector of indices of the n largest values in $\sigma(\mathbf{v})$, then the loss function for the second branch can be written as

$$L_{\text{reg}} = -\sum_{j=1}^{m} \tilde{\mathbf{w}}_j \log\left(\frac{e^{\mathbf{w}_j}}{\sum_j e^{\mathbf{w}_j}}\right) + \lambda ||\mathbf{W_2}||_2^2 \quad \forall j \in \Omega, \tag{10}$$

where $\mathbf{W_2}$ represents the weights of the final three layers of Branch 3 and $\tilde{\mathbf{w}}$ is the ground truth vector of weights for the given image. The entries of $\tilde{\mathbf{w}}$ are set based on the angular difference between the quaternion of the attitude class in question and the ground truth quaternion of the given image. Specifically,

$$\tilde{\mathbf{w}}_j = \frac{1 - \boldsymbol{\alpha}_j/\pi^2}{n - \sum_j \boldsymbol{\alpha}_j/\pi^2}, \forall j \in \Omega, \tag{11}$$

where $\boldsymbol{\alpha}_j$ is the angular difference between the unit quaternion representing the attitude class in question and the unit quaternion representing the ground truth of the relative attitude of the given image. Hence, the total loss function for relative attitude estimation can be written as:

$$L_{\text{total}} = L_{\text{class}} + \mu L_{\text{reg}} \tag{12}$$

The classification loss, L_{class}, is designed to force Branch 2's coefficients to be correlated with macroscopic changes in the relative attitude without penalizing classification predictions of attitude classes that have small angular difference between them. Instead, the regularization loss, L_{reg} is designed to force the Branch 3's coefficients to be correlated with the microscopic differences between adjacent attitude classes.

During inference, the estimate of the relative attitude, $\tilde{\mathbf{q}}_{\mathbf{BC}}$, is computed using an average of the unit quaternions of the n closest attitude classes, $\mathbf{Q}$, weighted by their corresponding relative weights, $\mathbf{\Gamma}$. This is akin to minimizing a weighted sum of the squared Frobenius norms of the differences between the rotation matrix representation of $\mathbf{Q}_i$ and $\mathbf{q}$

$$\tilde{\mathbf{q}}_{\mathbf{BC}} = \arg\min_{\mathbf{q} \in \mathbb{S}^3} \sum_i \mathbf{\Gamma}_i ||R(\mathbf{q}) - R(\mathbf{Q}_i)||_F^2 \tag{13}$$

where $\mathbb{S}^3$ denotes the unit 3-sphere.[34] The pseudo-code used to compute the weighted average is presented below as Algorithm 2.

SPACECRAFT POSE ESTIMATION DATASET (SPEED)

Training a convolutional neural network usually requires extremely large labeled image datasets such as ImageNet[35] and Places,[36] which contain millions of images. Collecting and labeling such amount of actual space imagery is extremely difficult. Therefore, this work introduces SPEED, a

Algorithm 2: Computes a unit quaternion that is an average of the input unit quaternions $\mathbf{Q}$ weighted by the input weights $\mathbf{\Gamma}$.

```
Input  : Matrix Q, Vector Γ
Output: Vector q
A ← zeros(4,4)
M ← length(Q,1)
foreach i in range(M) do
    $q$ = $\mathbf{Q}_i$
    $w_i$ = $\mathbf{\Gamma}_i$
    A = $w_i$ .* $(q * q')$ + A
    alpha = alpha + $w_i$
end
A = (1.0 / alpha) * A
// Get the eigenvector corresponding to the largest eigenvalue
q← eigs(A,1)
```

Table 1. Parameters of the camera used to capture the SPEED images.

Parameter	Description	Value
N_u	Number of horizontal pixels	1920
N_v	Number of vertical pixels	1200
f_x	Horizontal focal length	0.0176 m
f_y	Vertical focal length	0.0176 m
du	Horizontal pixel length	$5.86 \cdot 10^{-6}$ m
dv	Vertical pixel length	$5.86 \cdot 10^{-6}$ m

dataset of images that enables not only training and validation of the SPN method but also benchmarking various state-of-the-art monocular vision-based pose estimation techniques. The SPEED images and the corresponding ground truth pose information are generated using two key complementary sources at the Space Rendezvous Laboratory (SLAB) of Stanford University.[37] The first source produced 15000 augmented reality images based on the Optical Stimulator[17,38] camera emulator software and actual images of the Earth captured by the Himawari-8 geostationary meteorological satellite.[39] The second source produced 300 actual camera images of a 1:1 mock-up of the Tango spacecraft using the Testbed for Rendezvous and Optical Navigation (TRON). Table 1 provides the camera model used for both these sources.

The first source renders synthetic images of the Tango spacecraft using MATLAB and C++ language bindings of OpenGL. To create a diverse set of views of the target spacecraft, a set of relative attitudes and relative positions are selected. Unit quaternion parametrization of uniformly random rotations in the $SO(3)$ space are selected using Algorithm 1. Figure 10 shows the distribution of $\mathbf{q_{BC}}$ in the SPEED images, parametrized as Euler angles. The relative positions are obtained by separately selecting the relative distance and the bearing angles (defined in Equations 4 and 5). The bearing angles are uncorrelated random values selected from a multivariate normal distribution, $D_1 \sim \mathcal{N}(\boldsymbol{\mu} = [N_u/2,\ N_v/2]\text{px},\ \boldsymbol{\sigma} = [5N_u/2,\ 5N_v/2]\text{px})$. The relative distance is randomly selected from a standard normal distribution, $D_2 \sim \mathcal{N}(\mu = 3\text{m}, \sigma = 10\text{m})$. Any relative distance values below 3 meters and above 50 meters are rejected. Figure 7 shows the distribution of $\mathbf{t_{BC}}$ in the SPEED images.

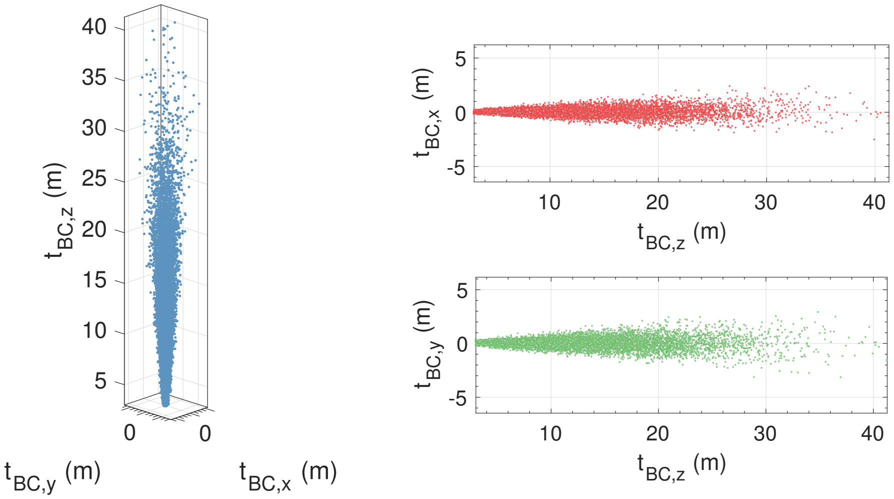

Figure 7. The distribution of the relative position, $\mathbf{t_{BC}}$, in the SPEED images.

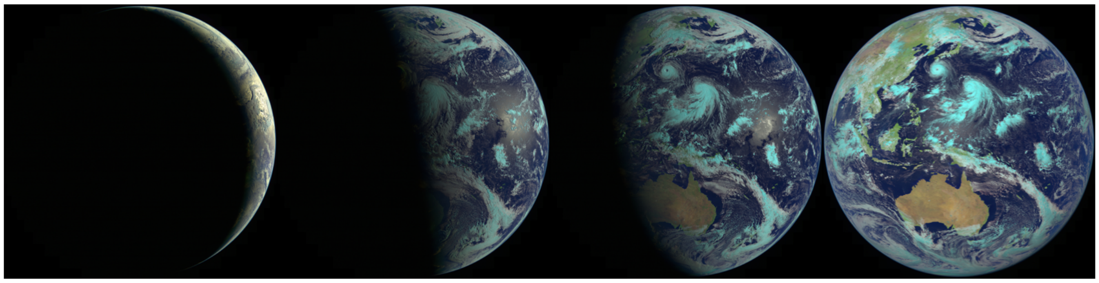

Figure 8. A montage of four of the 72 full-disk Earth images used to generate the background for half of the SPEED synthetic images.

The azimuth and elevation angles for the solar illumination are specifically chosen to match the solar illumination in the 72 actual images of the Earth captured by the Himawari-8 geostationary meteorological satellite. Figure 8 shows a montage of these images. The 72 images each provide a $100 \cdot 10^6$ pixels resolution disk-view of the Earth and were taken 10 minutes apart from each other over a period of 12 hours. Each of the images is converted to grayscale, cropped and inserted in as the background for half of the synthetic images of the Tango spacecraft. The location of the crop is selected at random from a uniform distribution spanning the Earth image. The size of the crop is selected to match the scale of the Earth when viewed through a camera (described in Table 1) located at an altitude of 700 km and pointed in the nadir direction. Gaussian blurring and white noise are added to all images to emulate the depth of field and shot noise, respectively. Figure 9 shows a montage of the resulting augmented reality images.

The second source of the SPEED images is the TRON facility at SLAB. It consists of a 7 degrees-of-freedom robotic arm, which positions and orients a vision-based sensor with respect to a target

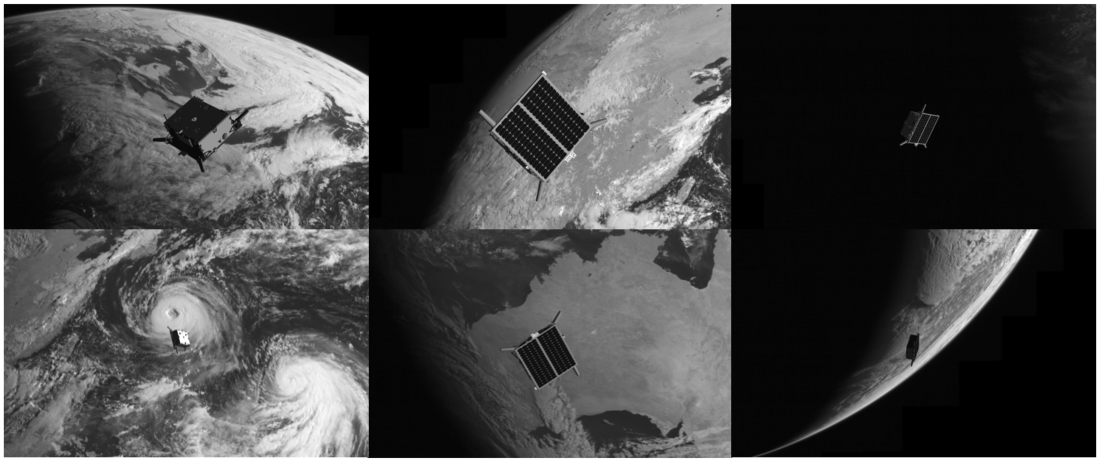

Figure 9. A montage of six synthetic images from the SPEED training-set.

object or scene. Custom illumination devices simulate Earth albedo and Sun light to high fidelity to emulate the illumination conditions present in space.[40] TRON provides images of a 1:1 mock-up model of the Tango spacecraft using a Point Grey Grasshopper 3 camera with a Xenoplan 1.9/17 mm lens. Note that this is the same camera as the one used in the first source. Calibrated motion capture cameras report the positions and attitudes of the camera and the Tango spacecraft, which are then used to calculate the ground truth pose of Tango with respect to the camera. Figure 11 shows a montage of the SPEED actual camera images.

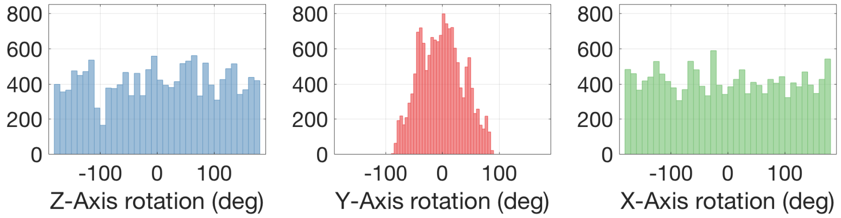

Figure 10. The distribution of the relative attitude in the SPEED images. For purposes of visualization, the relative attitude is parametrized as Euler angles.

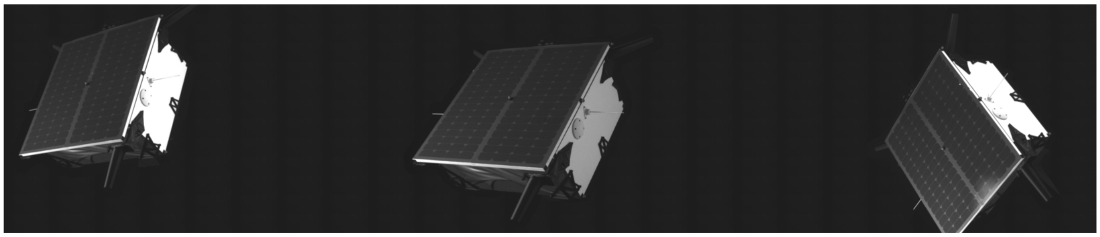

Figure 11. A montage of three actual camera images from the SPEED real test-set.

The SPEED training-set consists of 12000 synthetic images from the first source while the remainder 3000 synthetic images and the 300 actual camera images from the second source are available as two separate test-sets. The motivation for excluding the actual camera images from the training-set is to evaluate robustness and the domain adaptation capabilities of the pose estimation techniques.

EXPERIMENTS

The proposed SPN method was trained and tested using the SPEED images. For all experiments, the region proposal network, the convolutional layers, and the fully connected layers were pre-trained on the relatively larger ImageNet dataset.[35] Following that, the region proposal network and the fully connected layers were trained using 80% of the SPEED training-set while the remainder 20% was used for validation. The two sets of fully connected layers were trained jointly while the region proposal network was trained separately. Both training routines were carried out using stochastic gradient in batches of 16 images. The initial learning-rate was set to 0.003 with an exponential decay of 0.95 every 1000 steps. For attitude determination, the hyper-parameters m and n were set to 1000 and 5, respectively. During training, each image was resized to 224 $\times$ 224 pixels to match the input size of the Zeiler and Fergus model architecture.[32]

For quantitative analysis of the performance, three separate metrics are reported. To measure the accuracy of the 2D bounding box detection as compared with the ground truth 2D bounding box, the Intersection-Over-Union (IoU) metric is reported as

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}. \tag{14}$$

To measure the accuracy of the estimated relative position, $\tilde{\mathbf{t}}_{\mathbf{BC}}$ and the ground truth relative position, $\mathbf{t}_{\mathbf{BC}}$, the translation error for each image can be calculated as

$$E_T = |\mathbf{t}_{\mathbf{BC}} - \tilde{\mathbf{t}}_{\mathbf{BC}}| \tag{15}$$

To measure the accuracy of the estimated relative attitude, $\tilde{\mathbf{q}}_{\mathbf{BC}}$ and the ground truth relative attitude, $\mathbf{q}_{\mathbf{BC}}$, the attitude error for each image can be calculated as

$$E_R = 2\cos^{-1}(|z_s|), \text{ where} \tag{16}$$

$$\mathbf{z} = [z_s \ \mathbf{z_v}] = \mathbf{q}_{\mathbf{BC}} * \text{conj}(\tilde{\mathbf{q}}_{\mathbf{BC}}). \tag{17}$$

Table 2. Performance of the SPN pose estimation technique on the real and synthetic SPEED test-sets.

Metric	SPEED synthetic test-set	SPEED real test-set
Mean IoU (-)	0.8582	0.8596
Median IoU (-)	0.8908	0.8642
Mean E_T (m)	[0.055 0.046 0.78]	[0.036 0.015 0.189]
Median E_T (m)	[0.024 0.021 0.496]	[0.029 0.013 0.191]
Mean E_R (deg)	8.4254	18.188
Median E_R (deg)	7.0689	13.208

Table 2 lists the mean and median values of the three performance metrics for the SPEED synthetic and real test-sets. The levels of the relative position and attitude errors improve upon those of other pose estimation methods applied on similar imagery in previous work.[5,17] Notably, the 2D bounding box detection is less affected by the gap between the synthetic and real test-sets as compared with the attitude estimation. The E_T values for the actual camera images are lower than that of the synthetic images since the relative distances for the actual camera images is comparatively much lower than that of the synthetic images.

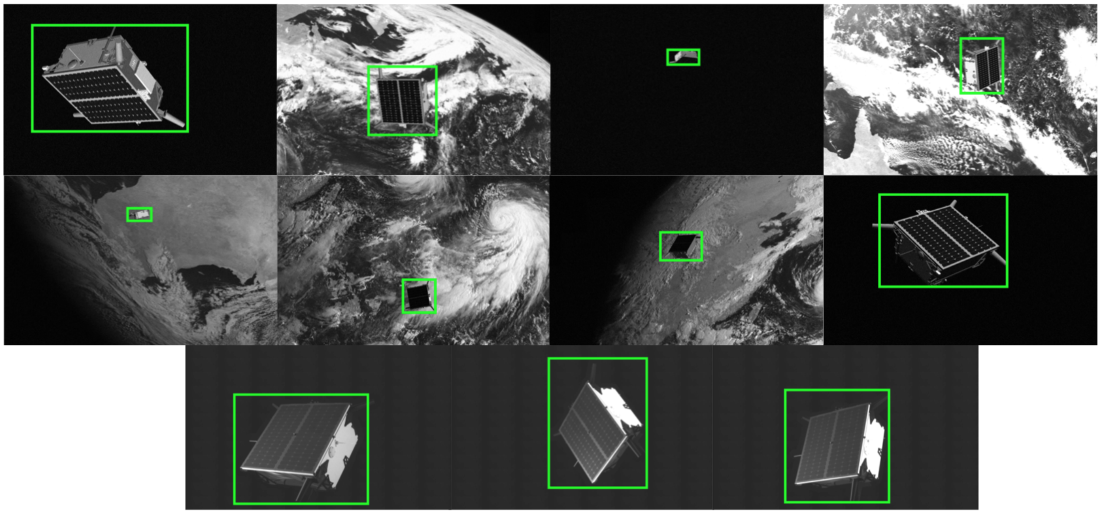

Figure 12. A montage of a few images with the 2D bounding box detections produced by the SPN method on the SPEED synthetic test-set.

Figure 12 shows the qualitative results of the 2D bounding box detection on the SPEED synthetic and real test-sets. The region proposal network used in the SPN method was successful in detecting the 2D bounding box in images regardless of whether the Earth is visible in the background. In general, the bounding box detection worked marginally better on actual camera images as compared to synthetic images since all actual camera images had a black background. The bounding box detection was successful even in cases with strong shadows as long as one side of the spacecraft was illuminated.

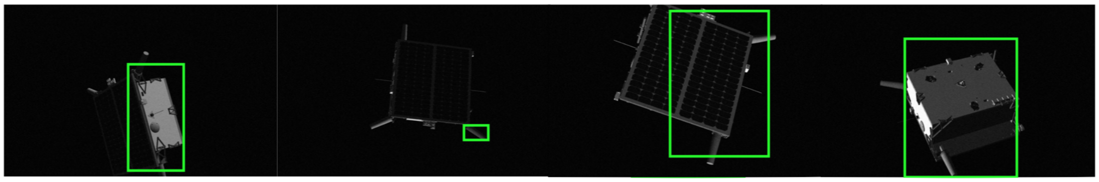

Figure 13. A montage of a few images from the SPEED synthetic test-set with inaccurate 2D bounding box detections.

However, the bounding box detection tends to fail in cases where the target spacecraft was in eclipse or when the target spacecraft was closer than ∼5 meters. Figure 13 shows a few examples from both these failure modes. At relative distances less than ∼5 meters, the Tango spacecraft starts getting clipped by the image boundaries leading to unpredictable behavior of the detection. In a docking scenario, this could potentially be resolved by performing pose estimation relative to the docking mechanism or another smaller fixture, instead of the entire spacecraft. Note that the SPN method can be extended to detect multiple 2D bounding boxes corresponding to the fixtures of interest and performing attitude estimation and position determination for each.

To further examine such trends, the three performance metrics are plotted against the relative distance. In particular, the images from the SPEED synthetic test-set were grouped into batches of 100 each according to their ground truth relative distance, $||\mathbf{t_{BC}}||_2$. The mean value of the performance metric was then plotted against the mean ground truth relative distance for each batch.

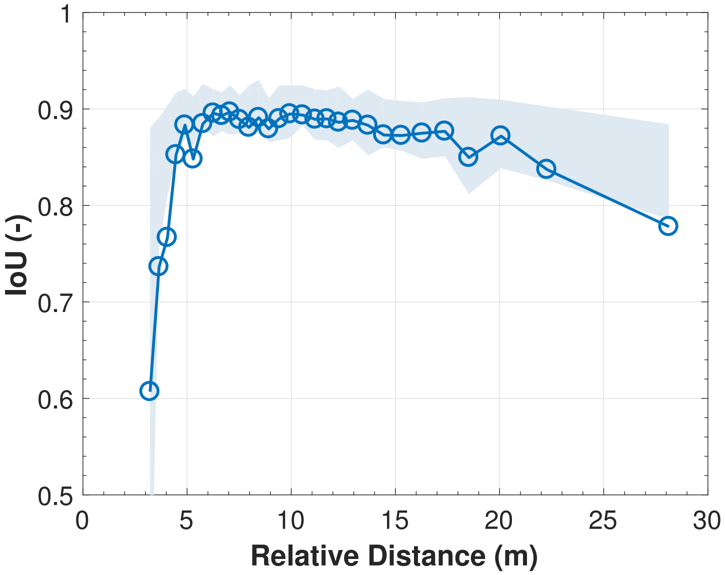

Figure 14. Mean IoU plotted against mean relative distance for the SPEED synthetic test-set. The shaded region shows the 25 and 75 percentile values.

Figure 14 shows the mean IoU values of the SPN method for the SPEED synthetic test-set plotted against the mean relative distance. The bounding box detection has the highest accuracy at ranges of ∼7 meters to ∼20 meters while there is a sharp drop-off in bounding box detection at relatively closer distances of less than ∼5 meters. In contrast, there is a gradual drop-off in performance at relatively farther distances as the target spacecraft starts occupying too few pixels in the image plane to allow for an accurate 2D bounding box detection. Degradation in performance at larger separations is also contributed to by the fact that the SPEED training-set contains more images at lower relative distances.

Figure 15 shows the mean E_R values of the SPN method for the SPEED synthetic test-set plotted against the mean relative distance. The mean E_R values are between 5 and 10 degrees for most of the relative distances. The range of the relative distances where attitude estimation has the highest accuracy is similar to the bounding box detection. This makes sense since both the region proposal network and the two sets of fully connected layers share the output of shared convolutional layers in the SPN method. Unlike the bounding box detection, the attitude estimation has a sharp drop-off in performance at both high and low relative distances. This is due to the presence of more outliers in the attitude estimation as compared with bounding box detection at high relative distances.

The estimated relative attitude value for each image in the SPEED synthetic test-set was then combined with the corresponding 2D bounding box detection to produce estimated values of the relative position. Figure 16 shows the three components of mean E_T values of the SPN method plotted against the mean relative distance. Note that the z-axis is aligned with the camera boresight direction while the x-axis and y-axis (lateral directions) are aligned with the image plane axes. The

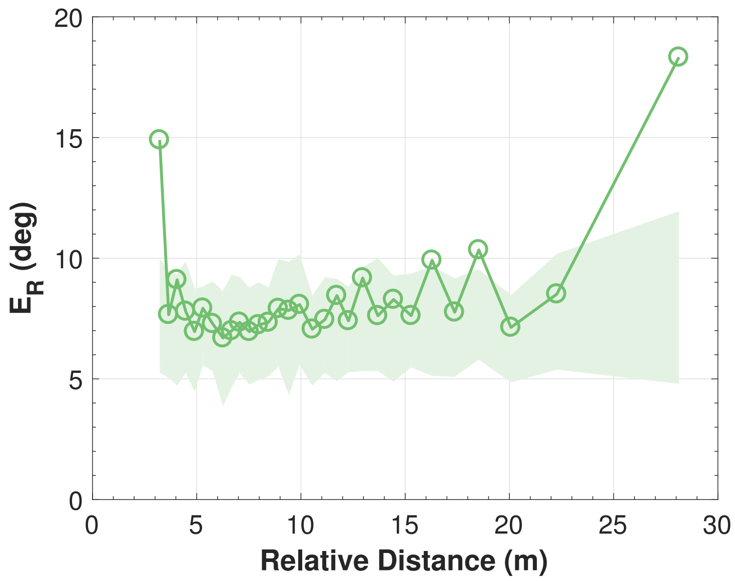

Figure 15. **Mean E_R plotted against mean relative distance for the SPEED synthetic test-set. The shaded region shows the 25 and 75 percentile values.**

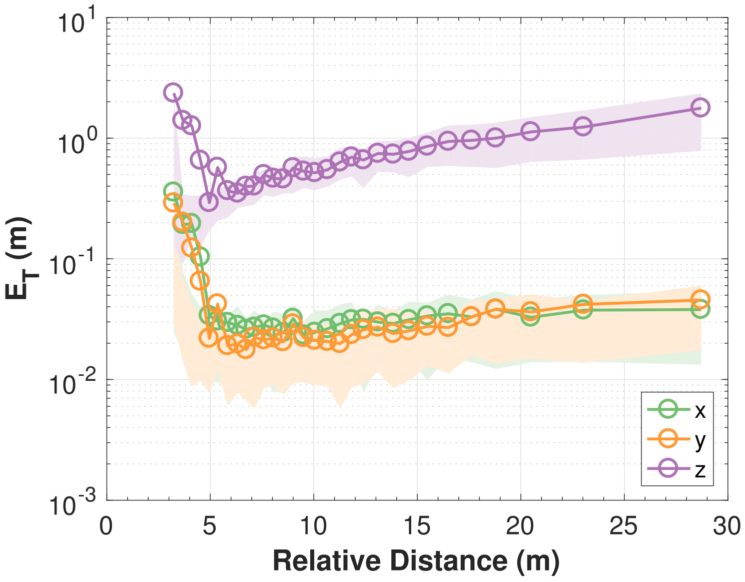

Figure 16. **Mean E_T plotted against mean relative distance for the SPEED synthetic test-set. The shaded region shows the 25 and 75 percentile values.**

errors in the lateral directions was an order of magnitude lower than the camera boresight direction, which implies that the bounding box detection was fairly successful at estimating the center of the bounding box as compared with the size of the bounding box. Trends in performance degradation of E_T mirror those observed for E_R and IoU.

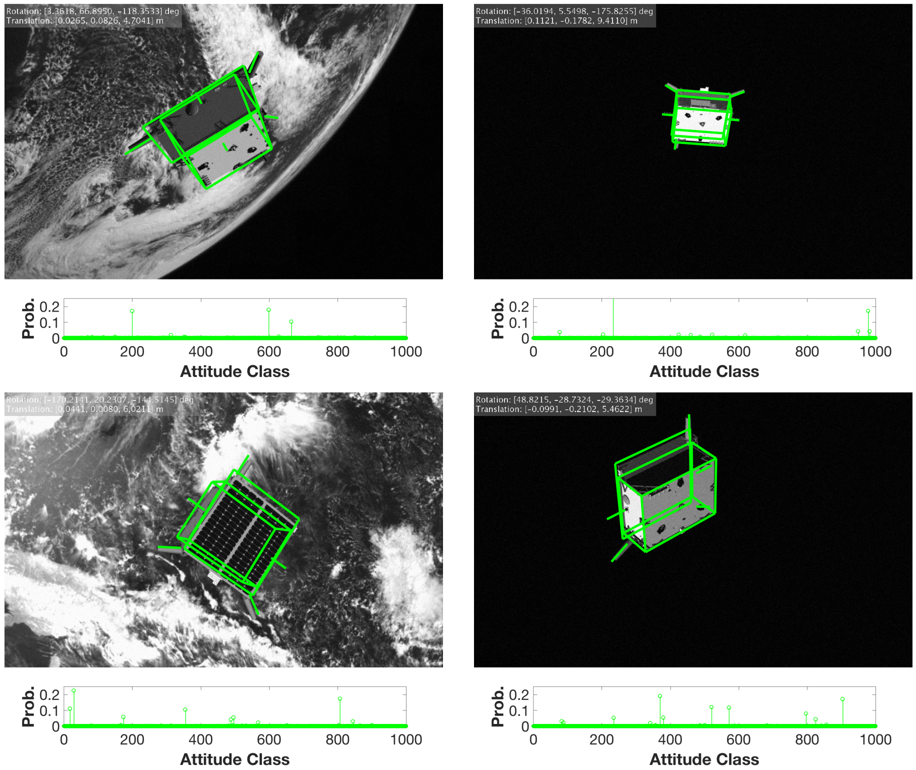

Figure 17. A montage of a few images with the pose solutions produced by the SPN method on the on the SPEED synthetic test-set.

Figure 17 and Figure 18 show some qualitative results of the pose estimated by the SPN method on the SPEED real and synthetic test-sets, respectively. The probability distribution output from the Branch 2 is also plotted for the respective images. In future, the probability distribution allows the setting up of a confidence metric to reject outliers in addition to being used by a navigation filter to accumulate information from sequential images to provide a more accurate estimate at a high rate. Predictably, the peaks in the probability distribution are lower for the SPEED real test-set as compared with the SPEED synthetic test-set since the convolutional neural network has overfit the synthetic images in the SPEED training-set to some extent. This could possibly be addressed by a stronger L2 regularization parameter and/or augmenting the training-set with images containing randomized textures of the target spacecraft similar to work in the area of domain adaptation[41,42] and domain randomization.[43,44]

CONCLUSIONS

This work introduces the SPN method to estimate the relative pose of a target spacecraft using a single grayscale image without requiring a-priori pose information. The SPN method makes the novel use of a hybrid classification and regression technique to estimate the relative attitude. The SPN method leverages the geometric knowledge of the perspective transformation and advances in

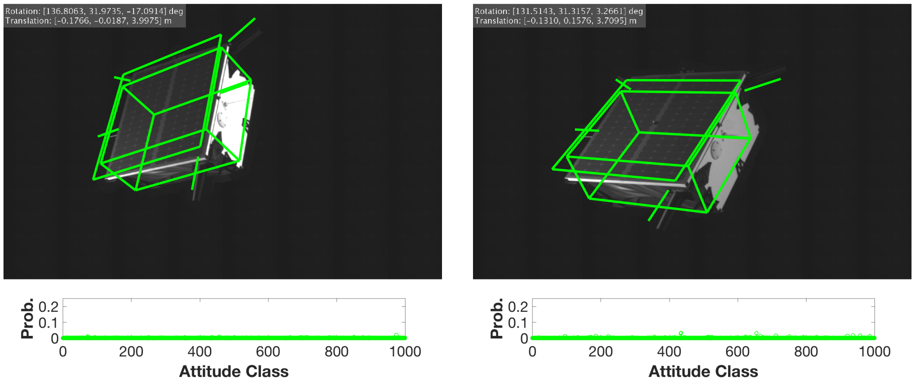
Rotation: [136.8063, 31.9735, -17.0914] deg
Translation: [-0.1766, -0.0187, 3.9975] m
Prob.
0.2
0.1
0
0
200
400
600
800
1000
Attitude Class
Rotation: [131.5143, 31.3157, 3.2661] deg
Translation: [-0.1310, 0.1576, 3.7095] m
Prob.
0.2
0.1
0
0
200
400
600
800
1000
Attitude Class

Figure 18. A montage of a few images with the pose solutions produced by the SPN method on the on the SPEED real test-set.

2D bounding box detection to estimate the relative position using the Gauss-Newton algorithm. This work also introduces SPEED, a publicly available dataset to allow for the training and validation of monocular pose estimation techniques. The SPEED training-set consists of 12000 augmented reality images created by fusing synthetic images of a target spacecraft with actual camera images of the Earth. Apart from the 3000 images of the Tango spacecraft in the SPEED synthetic test-set, 300 actual camera images of the same spacecraft are provided in the SPEED real test-set. The subsequent application of the SPN method on the SPEED synthetic test-set produces degree-level mean relative attitude errors and centimeter-level mean relative position errors exceeding the performance of conventional feature-based methods used in previous work. The pose estimation performance carried over to the actual camera images as well albeit with slightly higher errors due to the gap between the synthetic images used during training and actual camera images used for testing.

However, further work is required in a few directions. First, a complete assessment of how the SPN method stacks against conventional feature-based approaches as well as the more recent deep learning-based methods needs to be performed. The SPEED images and the associated performance metrics allow an excellent framework to carry out this assessment. Second, data augmentation techniques and/or stronger regularization during training is required to bridge the performance gap of the SPN method between the synthetic and real test-sets. Third, the performance drop-off at relative distances where the target spacecraft is only partially visible needs to be addressed to allow for pose estimation during all stages of close proximity operations. Lastly, the SPN method needs to be embedded in flight-grade hardware to profile its computational runtime and memory usage.

ACKNOWLEDGMENTS

The authors would like to thank the King Abdulaziz City for Science and Technology (KACST) Center of Excellence for research in Aeronautics & Astronautics (CEAA) at Stanford University for sponsoring this work. The authors would like to thank OHB Sweden, the German Aerospace Center (DLR), and the Technical University of Denmark (DTU) for the 3D model of the Tango spacecraft used to create the images used in this work. The authors would like to thank Nathan Stacey of the Space Rendezvous Laboratory at Stanford University for his technical contributions in generating ground truth pose information for the actual camera images of SPEED.

REFERENCES

[1] J. L. Forshaw, G. S. Aglietti, N. Navarathinam, H. Kadhem, T. Salmon, A. Pisseloup, E. Joffre, T. Chabot, I. Retat, R. Axthelm, S. Barraclough, A. Ratcliffe, C. Bernal, F. Chaumette, A. Pollini, and W. H. Steyn, "RemoveDEBRIS: An in-orbit active debris removal demonstration mission," *Acta Astronautica*, Vol. 127, No. 2016, 2016, pp. 448–463, 10.1016/j.actaastro.2016.06.018.

[2] B. Sullivan, D. Barnhart, L. Hill, P. Oppenheimer, B. L. Benedict, G. Van Ommering, L. Chappell, J. Ratti, and P. Will, "DARPA Phoenix Payload Orbital Delivery System (PODs): FedEx to GEO," *AIAA SPACE 2013 Conference and Exposition*, 2013, pp. 1–14, 10.2514/6.2013-5484.

[3] B. B. Reed, R. C. Smith, B. J. Naasz, J. F. Pellegrino, and C. E. Bacon, "The Restore-L Servicing Mission," *AIAA Space Forum*, Long Beach, CA, 2016, pp. 1–8, 10.2514/6.2016-5478.

[4] S. D'Amico, M. Benn, and J. Jorgensen, "Pose estimation of an uncooperative spacecraft from actual space imagery," *Proceedings of 5th International Conference on Spacecraft Formation Flying Missions and Technologies*, No. 1, 2013, pp. 1–17.

[5] S. Sharma, J. Ventura, and S. D'Amico, "Robust Model-Based Monocular Pose Initialization for Noncooperative Spacecraft Rendezvous," *Journal of Spacecraft and Rockets*, 2018, pp. 1–16, 10.2514/1.A34124.

[6] A. Cropp and P. Palmer, "Pose Estimation and Relative Orbit Determination of a Nearby Target Microsatellite using Passive Imagery," *5th Cranfield Conference on Dynamics and Control of Systems and Structures in Space 2002*, 2002, pp. 389–395.

[7] C. Liu and W. Hu, "Relative pose estimation for cylinder-shaped spacecrafts using single image," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 50, No. 4, 2014, pp. 3036–3056, 10.1109/TAES.2014.120757.

[8] B. J. Naasz, J. Van Eepoel, S. Z. Queen, C. M. Southward, and J. Hannah, "Flight results from the HST SM4 Relative Navigation Sensor system," *33rd Annual AAS Guidance and Control Conference*, Breckenridge, CO, USA, 2010.

[9] V. Capuano, G. Cuciniello, V. Pesce, R. Opromolla, S. Sarno, M. Lavagna, M. Grassi, F. Corraro, G. Capuano, P. Tabacco, F. Meta, M. L. Battagliere, and T. Alberto, "VINAG: A highly integrated system for autonomous on-board absolute and relative spacecraft navigation," *The 4S Symposium 2018*, No. 1, 2018.

[10] J. Kelsey, J. Byrne, M. Cosgrove, S. Seereeram, and R. Mehra, "Vision-based relative pose estimation for autonomous rendezvous and docking," *2006 IEEE Aerospace Conference*, 2006, 10.1109/AERO.2006.1655916.

[11] S. Sharma and S. D'Amico, "Comparative assessment of techniques for initial pose estimation using monocular vision," *Acta Astronautica*, Vol. 123, 2015, pp. 435–445, 10.1016/j.actaastro.2015.12.032.

[12] P. Lunghi, L. Losi, V. Pesce, and M. Lavagna, "Ground testing of vision-based GNC systems by means of a new experimental facility," *69th International Astronautical Congress (IAC)*, Bremen, Germany, IAF, 2018, pp. 1–15.

[13] A. Petit, E. Marchand, and K. Kanani, "Vision-based space autonomous rendezvous: A case study," *IEEE International Conference on Intelligent Robots and Systems*, 2011, pp. 619–624, 10.1109/IROS.2011.6048176.

[14] S. Zhang and X. Cao, "Closedform solution of monocular visionbased relative pose determination for RVD spacecrafts," *Aircraft Engineering and Aerospace Technology*, Vol. 77, No. 3, 2005, pp. 192–198, 10.1108/00022660510597214.

[15] M. Avilés, D. Mora, M. Canetri, and P. Colmenarejo, "A Complete IP-based Navigation Solution for the Approach and Capture of Active Debris," *67th International Astronautical Congress*, 2016, pp. 1–8.

[16] S. Sharma and S. D'Amico, "Reduced-Dynamics Pose Estimation for Non-Cooperative Spacecraft Rendezvous using Monocular Vision," *Proceedings of the 40th Annual AAS Rocky Mountain Section Guidance and Control Conference*, Breckenridge, CO, 2017, pp. 1–25.

[17] S. Sharma, C. Beierle, and S. D'Amico, "Pose Estimation for Non-Cooperative Spacecraft Rendezvous Using Convolutional Neural Networks," *2018 IEEE Aerospace Conference*, Big Sky, USA, IEEE, 2018, pp. 1–12.

[18] H. Su, C. R. Qi, Y. Li, and L. J. Guibas, "Render for CNN: Viewpoint estimation in images using CNNs trained with rendered 3D model views," *Proceedings of the IEEE International Conference on Computer Vision*, Vol. 11-18-Dece, 2016, pp. 2686–2694, 10.1109/ICCV.2015.308.

[19] S. Tulsiani and J. Malik, "Viewpoints and keypoints," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1510–1519.

[20] S. Sharma, C. Beierle, and S. D'Amico, "Towards Pose Determination for Non-Cooperative Spacecraft using Convolutional Neural Networks," *Proceedings of the 1st IAA Conference on Space Situational Awareness (ICSSA)*, 2017, pp. 1–5.

[21] S. Mahendran, H. Ali, and R. Vidal, “3D Pose Regression Using Convolutional Neural Networks,” *Proceedings - 2017 IEEE International Conference on Computer Vision Workshops, ICCVW 2017*, Vol. 2018-Janua, 2018, pp. 2174–2182, 10.1109/ICCVW.2017.254.

[22] A. Kendall, M. Grimes, and R. Cipolla, “PoseNet: A convolutional network for real-time 6-dof camera relocalization,” *Proceedings of the IEEE International Conference on Computer Vision*, Vol. 2015 Inter, 2015, pp. 2938–2946, 10.1109/ICCV.2015.336.

[23] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, “PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes,” 2017, 10.15607/RSS.2018.XIV.019.

[24] P. Besl and N. McKay, “A Method for Registration of 3-D Shapes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 2, 1992, pp. 239–256, 10.1109/34.121791.

[25] European Space Agency, “Kelvins - ESA’s Advanced Concepts Competition Website,” `https://kelvins.esa.int`. Accessed Januray 4, 2019.

[26] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” *Advances In Neural Information Processing Systems*, 2015, pp. 91–99, 10.1109/TPAMI.2016.2577031.

[27] M. Kaushal, B. S. Khehra, and A. Sharma, “Soft Computing based object detection and tracking approaches: State-of-the-Art survey,” *Applied Soft Computing Journal*, Vol. 70, 2018, pp. 423–464, 10.1016/j.asoc.2018.05.023.

[28] K. Hara, R. Vemulapalli, and Rama Chellappa, “Designing Deep Convolutional Neural Networks for Continuous Object Orientation Estimation,” *ArXiv:1702.01499*, 2017.

[29] J. Wu, *Robotic Object Pose Estimation with Deep Neural Networks*. PhD thesis, Massachusetts Institute of Technology, 2018.

[30] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox, “DeepIM: Deep Iterative Matching for 6D Pose Estimation,” *ArXiv:1804.00175*, 2018.

[31] R. Girshick, “Fast R-CNN,” *ArXiv:1504.08083*, apr 2015.

[32] M. D. Zeiler and R. Fergus, “Visualizing and Understanding Convolutional Networks,” *European Conference On Computer Vision*, 2014, pp. 818–833.

[33] K. Shoemake, “Uniform Random Rotations,” *Graphics Gems III (IBM Version)*, pp. 124–132, Elsevier, 1992.

[34] F. L. Markley, Y. Cheng, J. L. Crassidis, and Y. Oshman, “Averaging Quaternions,” *Journal of Guidance, Control, and Dynamics*, Vol. 30, jul 2007, pp. 1193–1197, 10.2514/1.28949.

[35] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, “ImageNet Large Scale Visual Recognition Challenge,” *International Journal of Computer Vision*, Vol. 115, No. 3, 2015, pp. 211–252, 10.1007/s11263-015-0816-y.

[36] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, “Learning Deep Features for Scene Recognition using Places Database,” *Advances in Neural Information Processing Systems*, 2014, pp. 487–495.

[37] Space Rendezvous Laboratory, Stanford University, “SLAB - Multi-satellite systems for unrivaled space science and exploration,” `https://slab.stanford.edu`. Accessed Januray 4, 2019.

[38] C. Beierle and Simone D’Amico, “Variable Magnification Optical Stimulator for Training and Validation of Spaceborne Vision-Based Navigation,” *Journal of Spacecraft and Rockets (In Print)*, 2018.

[39] K. Bessho, K. Date, M. Hayashi, A. Ikeda, T. Imai, H. Inoue, Y. Kumagai, T. Miyakawa, H. Murata, T. Ohno, A. Okuyama, R. Oyama, Y. Sasaki, Y. Shimazu, K. Shimoji, Y. Sumida, M. Suzuki, H. Taniguchi, H. Tsuchiyama, D. Uesawa, H. Yokota, and R. Yoshida, “An Introduction to Himawari-8/9- Japan’s New-Generation Geostationary Meteorological Satellites,” *Journal of the Meteorological Society of Japan*, Vol. 94, No. 2, 2016, pp. 151–183, 10.2151/jmsj.2016-009.

[40] S. Sharma, A. Koenig, and J. Sullivan, “Verification of Light-box Devices for Earth Albedo Simulation,” `https://damicos.people.stanford.edu/sites/g/files/sbiybj2226/f/tn2016_sharmakoenigsullivan.pdf`, 2018.

[41] J. Blitzer, M. Dredze, and F. Pereira, “Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification,” *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, Prague, Czech Republic, Association for Computational Linguistics, 2007, pp. 440–447, 10.1029/RS006i008p00787.

[42] H. Daumé, “Frustratingly Easy Domain Adaptation,” *ArXiv:0907.1815v1*, 2009, 10.1.1.110.2062.

[43] M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, J. Schneider, S. Sidor, J. Tobin, P. Welinder, L. Weng, and W. Zaremba, “Learning Dexterous In-Hand Manipulation,” *ArXiv:1808.00177v2*, 2018, pp. 1–27.

[44] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, “Domain randomization for transferring deep neural networks from simulation to the real world,” *IEEE International Conference on Intelligent Robots and Systems*, 2017, pp. 23–30, 10.1109/IROS.2017.8202133.