Self-Supervised Auxiliary Task Learning for Estimating Satellite Orientation

Klaus Okkelberg, Jacob Lucas, Trent Kyono, Michael Abercrombie
The Boeing Company
Justin Fletcher, Matthew Phelps
Odyssey Systems Consulting

Abstract

Understanding the orientation or pose of a satellite is critical for space domain awareness. Recent advancements in direct pose estimation using convolutional neural networks (CNNs) have motivated us to examine a data-driven, end-to-end solution for estimating satellite pose. In this work, we use a CNN to directly estimate the pointing angle of a resolved LEO object. To improve the generalization of this task to varying objects, we perform classification as an auxiliary task for regularization. Both supervised and self-supervised learning methods are explored. We show on a large synthetic dataset, that multi-task self-supervision can improve the primary task of pointing angle estimation and improves generalization to held-out objects.

1. INTRODUCTION

For space domain awareness, it is critical to know the pose, i.e. position and attitude (orientation), of satellites and other space objects. It is attractive to perform pose estimation using monocular vision-based systems since these function over a wide range of distances. Traditional computer vision methods for pose estimation [1, 2] construct mappings between an object image and either a 3D model or a database of 2D images or features. They rely on hand-crafted features that are not robust to image variations or across different objects. Deep learning-based methods [3–5] use neural networks that are trained to take an input image and either directly output the pose or output intermediate features that are used to regress the pose. These methods are more robust to image variations, but it is unclear whether their learned feature representations are generalizable.

Of the deep learning methods, those using convolutional neural networks (CNNs) are particularly attractive. CNNs are neural networks constructed using convolutional masks inspired by the human visual cortex. CNNs are robust to a wide range of degradations common in astronomical imagery, such as low resolution, low signal to noise ratio (SNR), harsh illumination, and atmospheric turbulence. However, training of CNNs requires a large number of labeled images to prevent overfitting and increase generalization. Generalizability refers to the performance difference of a model when evaluated on previously seen data (training data) versus new data (test data). Since large, high-quality datasets of satellites are difficult to obtain, other forms of regularization are required to prevent overfitting.

A common regularization method is to augment the training data with transforms of the images, including translation, rotation, flipping, blurring, brightness and contrast, and affine transforms. Augmentation increases the effective size of the training data, which reduces overfitting, and increases robustness to image variations. Another regularization method is introduce auxiliary tasks to the deep learning task, where the sole objective of the auxiliary tasks is to improve the performance of one or more primary tasks. This contrasts with multi-task learning where all the tasks are useful. The motivation is that the auxiliary tasks will lead to more robust and meaningful representations of image in the shared neural network layers by preventing trivial representations.

This paper studies the use of an auxiliary classification task to regularize the primary task of orientation estimation of Low Earth Orbit (LEO) satellites from ground-based imaging. Both supervised learning, where the classification is known, and self-supervised learning, where the classification is unknown, are studied. We used synthetic images of six different satellites with unique poses, generated as they would appear viewed in the absense of atmosphere.

The key contributions of this paper are:

1. Multi-task learning with an auxiliary classification task improves the primary task of orientation estimation from single-frame satellite images and improves generalization to new data.

DISTRIBUTION A. Approved for public release: distribution is unlimited.
Public Affairs release approval #AFRL-2021-2922

Copyright © 2021 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

2. Class labels learned using semi-supervised learning can be used for the classification task when the actual classes are unknown and similarly increases performance.
3. A loss using the geodesic distance results in lower mean orientation estimation error than using the mean-squared error (MSE) for the 6D orientation representation in [6].

2. RELATED WORK

Satellite pose estimation. Classical satellite pose estimation [7–9] would use extracted features from a 2D image to iteratively find the best pose solution that minimizes some error criterion in the presence of outliers. Recent advancements in the design and training of deep neural networks have produced CNNs that either directly predict the pose or produce intermediate features that can be used to compute the pose. For example for direct pose estimation, PoseCNN [3] and the network in [10] directly regress the pose, and SPN [11] and the network in [12] classify the pose into a finite number of bins. For 2D feature extraction, KPD [13] and PVNet [14] have been used. These 2D keypoints are then correlated with those on 3D models, and a Perspective-n-Point (PnP) solver is used to predict the pose [15].

Auxiliary Task Learning. Multi-task learning with auxiliary tasks can be performed with shared hidden layers and task-specific output layers [16], which has been shown to reduce overfitting [17]. Each task can also have its own hidden layers, and regularization is used to encourage similarity among parameters [18, 19]. Recent works have focused on how related the auxiliary tasks need to be to the primary tasks and how the loss function should be weighted between primary and auxiliary tasks [20–22].

Self-Supervision. In self-supervised learning, the annotation of the data is automated. Example annotation methods include: image reconstruction through colorization [23], super-resolution [24], and in-painting [25]; pattern sensing through solving jigsaw puzzle [26], context prediction [27], and geometric transformations [28]; and automated label generation by using a procedural synthetic image generator or through clustering [29–31].

3. METHODS

3.1 Rigid-Body Orientation Estimation

The problem of 6D pose estimation involves finding the 3D translation and 3D rotation that transforms from the object coordinate system to the camera coordinate system. From the point-of-view of ground-based imaging, it can be assumed the translation is known via two-line element (TLE) or state-vector, which is known ahead of time or can be recovered from tracking the object. Therefore, in this work we focus only on estimating the rotation. Furthermore, we assume the input is a single image.

We use a 6D orientation representation that consists of the first two columns of the rotation matrix. The authors in [6] found that this produces a continuous representation of the rotation in $SO(3)$ that is beneficial for regression using neural networks. For reconstruction of the rotation matrix from the 6D form, we follow [6] and use a Gram-Schmidt-like process to produce an orthonormal matrix. Specifically, we reshape the 6D output to two 3D vectors, apply the modified Gram-Schmidt algorithm [32] to produce the first two orthonormal columns of the rotation matrix, and find the last column as the cross product of the first two.

For the loss function, the MSE between the 6D neural network output and the 6D form of the true orientation has been used. We feel that this loss function is overly constrained since for accurate estimation of the orientation using our reconstruction method, it is not necessary for the MSE to be minimal. An over constrained loss function can lead to overfitting on the training data. This paper explored using the geodesic distance between the reconstructed rotation matrix and the true rotation matrix as the loss function, defined as

$$\theta = \cos^{-1}\left(\frac{\mathrm{trace}(PQ^{\mathrm{T}}) - 1}{2}\right), \tag{1}$$

where θ is the minimum angular difference between the two rotations, P is the true rotation matrix, Q is the reconstructed rotation matrix, and $(\cdot)^{\mathrm{T}}$ is the matrix transpose. The geodesic error is also our performance metric, since it is independent of the orientation representation and most accuractely characterizes the angular error in estimating the orientation. For training, we used the geodesic error in radians, though performance results are reported in degrees.

DISTRIBUTION A. Approved for public release: distribution is unlimited.
Public Affairs release approval #AFRL-2021-2922

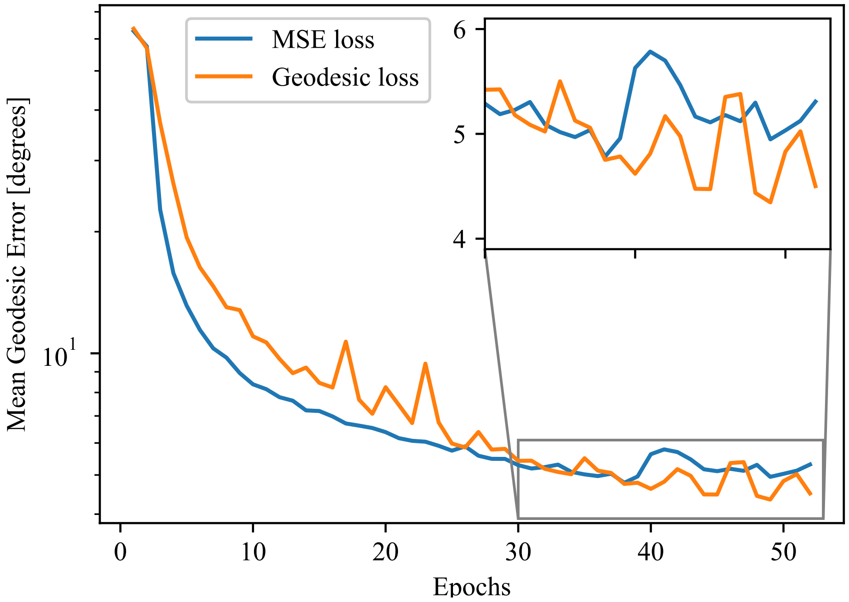

Fig. 1: Test performance of different loss functions for orientation estimation. Angular error using geodesic distance as loss converges slower than using MSE but achieves lower value. With the MSE loss, an increase in test error around epoch 40 shows that the model is overfitting.

We ran a short experiment to test the feasibility of the two candidate loss functions. The network was trained on the orientation estimation task for 52 epochs. In the first two epochs, the orientation layers were trained using MSE loss in both cases. The remaining 50 epochs trained the whole network using either MSE or geodesic distance as the loss. The AdamW optimizer [33] was used with a learning rate of 10^{-4}, a momentum of 0.9, and a weight decay of 10^{-2}. For evaluation of the performance, we calculated the geodesic error between estimated and true orientations on the test set of held-out images. The training results are shown in Fig. 1. The initial test performance is better with the MSE loss but overfitting starts occuring after 40 epochs, which can be seen by an increase in the angular error. The geodesic distance loss showed no overfitting and reached a lower error.

For the neural network architecture, we adapted a state-of-the-art CNN for image classification called Xception [34] that was trained on ImageNet. After the global average pooling layer in Xception, the linear regression layer used for classification was replaced by two sets of fully connected layer, batch normalization layer, and ReLU activation, followed by a final fully connected layer with 6 neurons, corresponding to the 6D orientation representation. A similar choice of pretrained network and additions was found by have good pose estimation performance in [10].

3.2 Auxiliary Task Learning

The goal of multi-task learning is to find a common feature representation in the initial shared layers of the neural network, while the individual tasks are completed using independent branches. This is akin to an encoder-decoder architecture in which after a common encoder, there is a specialized decoder for each task. While each task favors the learning of different features in the shared section, some features can be exploited by other tasks as well.

In auxiliary task learning, we separate the tasks into primary tasks and auxiliary tasks, which are of little to no interest. Though not directly related to the primary tasks, auxiliary tasks assist in the learning of more robust and meaningful representations in the shared layers. Auxiliary tasks should be easy to learn and uncorrelated with the primary tasks, ideally a global description of the image. They are a form of regularization that restricts the optimization parameter space by forcing the network to generalize to multiple tasks.

To complement the primary task of orientation estimation, we use classification of the satellite images as an auxiliary task. This is a global task that should be easy for the base Xception network that was designed for ImageNet, which has 1000 classes rather than the six classes in our dataset. For the network architecture, rather than directly regressing the

DISTRIBUTION A. Approved for public release: distribution is unlimited.
Public Affairs release approval #AFRL-2021-2922

features after the global average pooling layer as in typical image classification CNNs, we instead use a similar set of layers as for the orientation estimation task. The purpose of this is to reduce the regularization effect, since the two tasks are highly uncorrelated. This is also beneficial for increasing robustness to mislabeled images, which can occur with a self-supervised approach as discussed in the next section. After the final layer, a SoftMax layer is used to output probabilities for each class. We used cross-entropy as the loss function during training and accuracy as the performance metric. The total loss function for training was the unweighted sum of the orientation and classification losses.

3.3 Self-Supervised Classification

To increase the versability and viability of our auxiliary task learning method to data with missing or inaccurate labels, we use a self-supervised approach for labeling the satellite classes. Typical self-supervised methods use a pretext task to generate surrogate labels, turning an unsupervised learning problem into a supervised one, but these tasks are domain-dependent. Instead, we use a generalizable deep clustering approach to self-supervision as in [29], in which they iteratively cluster deep features from their CNN to produce pseudo-labels that are then used for supervised classification.

We use a similar clustering method, using the deep features after the common network layers. Principle component analysis (PCA) is applied to the features to reduce their dimensionality. K-means clustering of the ℓ_2-normalized PCA components produces the pseudo-labels, which are then used for the auxiliary classification task. To minimize training time, we chose to use 6 clusters to cover the 6 actual classes, though more clusters would tend to perform better as though would capture inter-class correlations. The initial clusters are computed by performing a forward pass of the unaugmented training data on the Xception network with pretrained ImageNet weights. For subsequent clustering, we use augmented training data to save training time. We found that recomputing the clusters every 10 epochs was a good tradeoff between increased training time versus regularization strength of the self-supervised auxiliary task.

4. EXPERIMENTS

For repeatability, all neural networks were trained using Python 3.7 and PyTorch 1.7 on NVidia Tesla V100 GPUs with 32 GB of VRAM. The operating system was Ubuntu Linux 18.

The data consisted of unique poses of six satellites rendered as if viewed without degradation from the Advanced Electro-Optical System (AEOS) at the summit of Haleakala, with approximately 10,000 images per class. The training images were augmented with a proprietary set of transforms. For testing, the images are unaugmented. The held-out testing images accounts for 10% of the total data.

All experiments used a batch size of 128 and were trained for 202 epochs. The optimizer used was AdamW [33] with Lookahead [35] with a weight decay of 10^{-4}. For the first two epochs, the MSE loss was used for the orientation estimation with a learning rate of 10^{-4} and momentum of 0.9. For the remaining epochs, the learning rate was set according to a “1cycle” policy [36] that was found to have faster convergence and better generalization than other learning schedules. Over the first 30% of the epochs, a cosine annealing strategy was used to vary the learning rate from 4×10^{-5} to 10^{-3} and the momentum from 0.85 to 0.95. For the remaining epochs, the learning rate was decreased from 10^{-3} to 10^{-7} and the momentum from 0.95 to 0.85 with a cosine annealing strategy.

We tested combinations of the MSE and geodesic error loss functions with whether an auxiliary task was used. Based on the results, we additionally tested a self-supervised auxiliary task with the geodesic distance loss. A summary of the experimental results is shown in Table 1. Using the geodesic distance loss with supervised auxiliary learning results in the lowest mean geodesic error, while using the MSE loss without an auxiliary task results in the lowest standard deviation. The classification accuracy, where applicable, was high for all experiments.

4.1 Supervised Auxiliary Task Learning

Fig. 2 shows the test performance for orientation estimation for both single-task and multi-task supervised learning with MSE and geodesic distance loss functions. For the single-task case, using the geodesic distance as the loss slightly outperforms using the MSE. With the auxiliary classification task, the performance with the geodesic loss is improved from the single-task case. The loss during training is also more stable with smaller and fewer spikes in error, showing the regularizing effect of the auxiliary task. For the MSE loss with the auxiliary task, there is a gap in the error as early as 10 epochs into the training process. This combination needs about 50 more epochs to achieve the same performance as the other cases.

**DISTRIBUTION A. Approved for public release: distribution is unlimited.
Public Affairs release approval #AFRL-2021-2922**

Table 1: Performance of various combinations of orientation loss functions and auxiliary tasks. Bold in each column indicates best value.

		Geo. error, mean [deg.]		Geo. error, std. dev. [deg.]		Class. acc. [%]	
Orient. loss	Aux. class.	Training	Test	Training	Test	Training	Test
MSE	None	1.36	2.02	**2.60**	**7.83**	—	—
MSE	Supervised	2.29	2.91	4.28	8.51	**99.99**	**99.92**
Geodesic	None	1.27	1.94	6.58	10.15	—	—
Geodesic	Supervised	**1.19**	**1.80**	5.82	9.46	**99.99**	99.90
Geodesic	Self-supervised	1.32	1.89	6.26	8.89	99.32*	98.78*

*Classification accuracy reported for deep clusters from self-supervision rather than actual labels.

From Table 1, the generalization gap (difference between test and training error) is lower when the auxiliary task is used. The table also shows that the standard deviation of the geodesic error is lowest for the MSE loss without an auxiliary task followed by the MSE loss with such a task, while those with the geodesic distances losses are significantly greater. Like with the mean error metric, with an auxiliary task the standard deviation of the error is increased with the MSE loss and decreased with the geodesic distance loss. The generalization gap for the standard deviation of the geodesic error is approximately lowest for the two geodesic distance loss cases, while the MSE loss without an auxiliary task has the largest gap. These standard deviation results show that the MSE is better for minimizing the mean-squared geodesic error, while using the geodesic distance as the loss minimizes the mean geodesic error. This suggests that if the target performance metric is to minimize the mean-squared geodesic error, the loss function should be the mean-squared geodesic distance or the root-mean-square geodesic distance.

The results show that the features learned for the classification task are incompatible with those necessary for orientation estimation with our network architecture when using the MSE as the loss function for orientation estimation, leading to a slower convergence rate and lower overall performance when the MSE loss is used for the auxiliary learning case. On the other hand, the classification features are beneficial when the geodesic loss is used, producing better generalization and better performance with auxiliary learning. The results suggest that there is a highly nonlinear relationship between the features important for classification and those important for estimating the values of the 6D orientation representation, which the linear regression layers are not able to find easily when using the MSE as the loss. However, with the geodesic loss, the orientation branch is able to take advantage of the Gram-Schmidt-like reconstruction method for the rotation matrix that reduces the nonlinearity between the two tasks. In effect, the neural network is not doing the orthonormalizing of the orientation representation, so it has additional capacity which allows it to more accurately estimate the orientation.

4.2 Self-Supverised Auxiliary Task Learning

Fig. 3 compares the performance using self-supervised classification for the auxiliary classification task to using supervised classification for the auxiliary task and using no auxiliary task. The results show that self-supervision increased performance versus the single-task case. The performance curve also has the fewest and smallest deviations, showing that self-supversion has a stronger regularization effect than regular supervision. However, it achieved worse performance than the supervised learning case, which suggests that mislabelings from the self-supervision process have a significant detrimental effect on performance. This effect can likely be reduced by clustering on unaugmented data, performing the deep clustering more frequently, using more clusters, or carrying out the clustering in a nonlinear projection space.

Table 1 shows that the self-supervised case has the smallest generalization gap for both the mean and the standard deviation of the geodesic error. This suggests that the periodic clustering in our self-supervision method minimizes overfitting in the neural network. The self-supervised case also has the lowest test geodesic error standard deviation of the geodesic distance loss cases, again showing the generalizability of this method.

DISTRIBUTION A. Approved for public release: distribution is unlimited.
Public Affairs release approval #AFRL-2021-2922

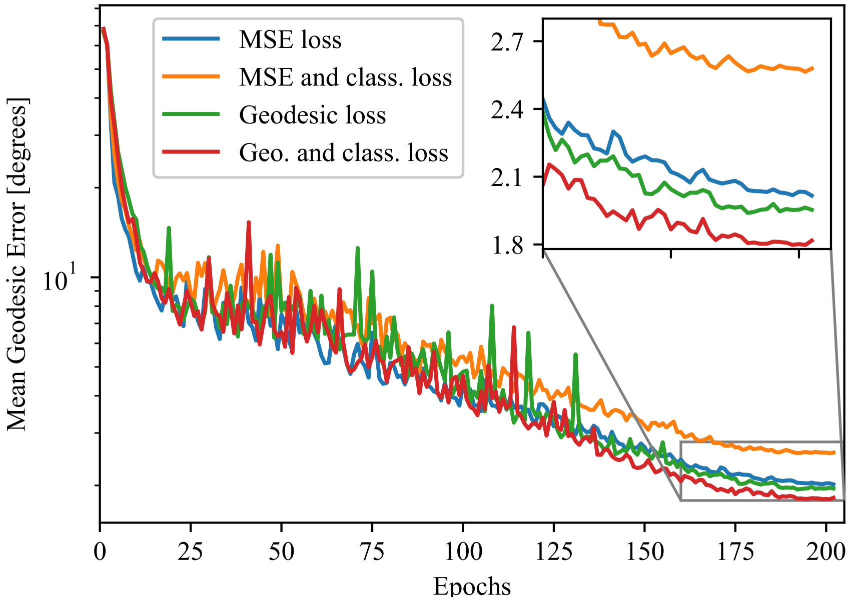

Fig. 2: Test performance for networks trained with different loss functions. Geodesic loss outperforms MSE loss. Auxiliary classification task helps performance with geodesic loss but worsens performance with MSE loss.

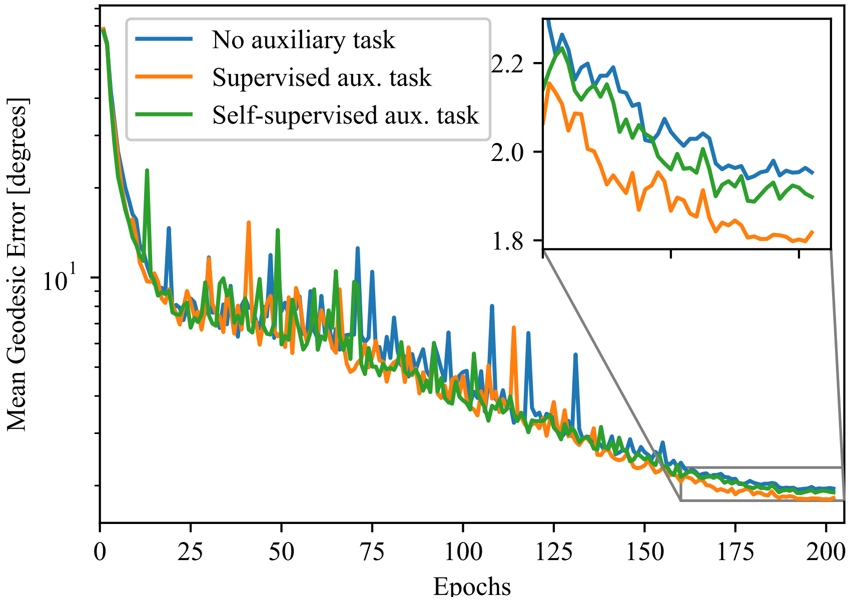

Fig. 3: Test performance for networks trained with various auxiliary tasks and geodesic distance for the orientation loss. Using an auxiliary task improves performance of the network. The improvement is greater with supervision versus self-supervision, because there are no mislabelings that could be learned through the self-supervised process.

DISTRIBUTION A. Approved for public release: distribution is unlimited.
Public Affairs release approval #AFRL-2021-2922

Copyright © 2021 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

5. CONCLUSION

This paper showed that orientation estimation of satellites using a CNN is affected by the choice of loss function, the use of an auxiliary classification task, and by whether the class labels are known or derived through self-supervision. We found that using geodesic distance as the loss function decreases the mean geodesic error. Further research is needed to find the best loss function for decreasing its standard deviation. We also observed that using an auxiliary classification task reduced the generalization gap. The auxiliary task also improved performance with the geodesic distance loss. Finally, we discovered that a self-supervised classification task improved performance, though not as much as with supervised classification. The generalization gap was minimal with self-supervision.

For future work, we plan on increasing the number of satellite classes, exploring other loss functions for orientation estimation, testing other auxiliary tasks such as image segmentation, adjusting the weight for the auxiliary task dynamically with a learned weight, and incorporating real data.

REFERENCES

[1] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, 1150–1157 vol.2, 1999. DOI: `10.1109/ICCV.1999.790410`.

[2] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes. In *Computer Vision – ACCV 2012*, pages 548–562, 2013. ISBN: 978-3-642-37331-2.

[3] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. PoseCNN: a convolutional neural network for 6D object pose estimation in cluttered scenes, 2018. arXiv: `1711.00199 [cs.CV]`.

[4] Thanh-Toan Do, Ming Cai, Trung Pham, and Ian Reid. Deep-6DPose: recovering 6D object pose from a single rgb image, 2018. arXiv: `1802.10367 [cs.CV]`.

[5] Georgios Pavlakos, Xiaowei Zhou, Aaron Chan, Konstantinos G. Derpanis, and Kostas Daniilidis. 6-DoF object pose from semantic keypoints. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2011–2018, 2017. DOI: `10.1109/ICRA.2017.7989233`.

[6] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5738–5746, 2019. DOI: `10.1109/CVPR.2019.00589`.

[7] A. Cropp and P. Palmer. Pose estimation and relative orbit determinationof a nearby target microsatellite using passive image. In *5th Cranfield Conference on Dynamics and Control of Systems and Structures in Space 2002*, 2002.

[8] Simone D'Amico, Mathias Benn, and John L. Jørgensen. Pose estimation of an uncooperative spacecraft from actual space imagery. *International Journal of Space Science and Engineering*, 2(2):171–189, 2014. DOI: `10.1504/IJSPACESE.2014.060600`.

[9] S. Zhang and X. Cao. Closed-form solution of monocular vision-based relative pose determination for rvd spacecrafts. *Aircraft Engineering and Aerospace Technology*, 77:192–198, 2005.

[10] Jacob Lucas, Trent Kyono, Michael Werth, Nicole Gagnier, Zackary Endsley, Ian Mcquaid, and Justin Fletcher. Estimating satellite orientation through turbulence with deep learning. In *Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, September 2020.

[11] Sumant Sharma, Connor Beierle, and Simone D'Amico. Pose estimation for non-cooperative spacecraft rendezvous using convolutional neural networks. In *2018 IEEE Aerospace Conference*, pages 1–12, 2018. DOI: `10.1109/AERO.2018.8396425`.

[12] Pedro F. Proença and Yang Gao. Deep learning for spacecraft pose estimation from photorealistic rendering. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6007–6013, 2020. DOI: `10.1109/ICRA40945.2020.9197244`.

[13] Zelin Zhao, Gao Peng, Haoyu Wang, Hao-Shu Fang, Chengkun Li, and Cewu Lu. Estimating 6D pose from localizing designated surface keypoints, 2018. arXiv: `1812.01387 [cs.CV]`.

DISTRIBUTION A. Approved for public release: distribution is unlimited.
Public Affairs release approval #AFRL-2021-2922

Copyright © 2021 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com

[14] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. PVNet: pixel-wise voting network for 6DoF pose estimation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4556–4565, 2019. DOI: `10.1109/CVPR.2019.00469`.

[15] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. EPnP: an accurate O(n) solution to the PnP problem. *Int. J. Comput. Vision*, 81(2):155–166, 2009. ISSN: 0920-5691. DOI: `10.1007/s11263-008-0152-6`.

[16] Richard Caruana. Multitask learning: a knowledge-based source of inductive bias. In *Proceedings of the Tenth International Conference on Machine Learning*, pages 41–48. Morgan Kaufmann, 1993.

[17] Jonathan Baxter. A bayesian/information theoretic model of learning to learn via multiple task sampling. In *Machine Learning*, pages 7–39, 1997.

[18] Long Duong, Trevor Cohn, Steven Bird, and Paul Cook. Low resource dependency parsing: cross-lingual parameter sharing in a neural network parser. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 845–850. Association for Computational Linguistics, July 2015. DOI: `10.3115/v1/P15-2139`.

[19] Yongxin Yang and Timothy M. Hospedales. Trace norm regularised deep multi-task learning, 2017. arXiv: `1606.04038 [cs.LG]`.

[20] Bernardino Romera Paredes, Andreas Argyriou, Nadia Berthouze, and Massimiliano Pontil. Exploiting unrelated tasks in multi-task learning. In *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, volume 22 of *Proceedings of Machine Learning Research*, pages 951–959, 2012.

[21] Roberto Cipolla, Yarin Gal, and Alex Kendall. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7482–7491, 2018. DOI: `10.1109/CVPR.2018.00781`.

[22] Lukas Liebel and Marco Körner. Auxiliary tasks in multi-task learning, 2018. arXiv: `1805.06334 [cs.CV]`.

[23] Richard Zhang, Jun-Yan Zhu, Phillip Isola, Xinyang Geng, Angela S. Lin, Tianhe Yu, and Alexei A. Efros. Real-time user-guided image colorization with learned deep priors. *ACM Trans. Graph.*, 36(4), July 2017. ISSN: 0730-0301. DOI: `10.1145/3072959.3073703`.

[24] Christian Ledig, Lucas Theis, Ferenc HuszÃ¡r, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 105–114, 2017. DOI: `10.1109/CVPR.2017.19`.

[25] Deepak Pathak, Philipp KrÃ€henbÃŒhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: feature learning by inpainting. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2536–2544, 2016. DOI: `10.1109/CVPR.2016.278`.

[26] Chen Wei, Lingxi Xie, Xutong Ren, Yingda Xia, Chi Su, Jiaying Liu, Qi Tian, and Alan L. Yuille. Iterative reorganization with weak spatial constraints: solving arbitrary jigsaw puzzles for unsupervised representation learning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1910–1919, 2019. DOI: `10.1109/CVPR.2019.00201`.

[27] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1422–1430, 2015. DOI: `10.1109/ICCV.2015.167`.

[28] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018. URL: `https://openreview.net/forum?id=S1v4N2l0-`.

[29] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features, 2019. arXiv: `1807.05520 [cs.CV]`.

[30] Yuki M. Asano, C. Rupprecht, and A. Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *International Conference on Learning Representations*, 2020. URL: `https://openreview.net/forum?id=Hyx-jyBFPr`.

[31] Miguel A. Bautista, Artsiom Sanakoyeu, Ekaterina Sutter, and Björn Ommer. Cliquecnn: deep unsupervised exemplar learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 3853–3861, 2016. ISBN: 9781510838819.

[32] Å. Björck. Numerics of Gram-Schmidt orthogonalization. *Linear Algebra and its Applications*, 197–198:297–316, 1994. ISSN: 0024-3795. DOI: https://doi.org/10.1016/0024-3795(94)90493-6. URL: https://www.sciencedirect.com/science/article/pii/0024379594904936.

[33] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. arXiv: 1711.05101 [cs.LG].

[34] François Chollet. Xception: deep learning with depthwise separable convolutions. *CoRR*, abs/1610.02357, 2016. arXiv: 1610.02357. URL: http://arxiv.org/abs/1610.02357.

[35] Michael Ruogu Zhang, James Lucas, Geoffrey E. Hinton, and Jimmy Ba. Lookahead optimizer: k steps forward, 1 step back. In *Neural Information Processing Systems*, 2019.

[36] Leslie N. Smith and Nicholay Topin. Super-convergence: very fast training of neural networks using large learning rates, 2018. arXiv: 1708.07120 [cs.LG].

Copyright © 2021 Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS) – www.amostech.com