Incremental Learning of Novel Resident Space Object Spectral Fingerprints

J. Zachary Gazak
United States Space Force
Ian McQuaid
Air Force Research Laboratory

Brandon Wolfson
Centauri

Justin Fletcher
United States Space Force

ABSTRACT

Positive identification of resident space objects (RSOs) injects rich information to traditionally position-only space traffic catalogs. Prior simulation work has demonstrated that convolutional neural network classifiers trained on spectroscopic data (SpectraNets) are highly effective, with accuracies on static 64 class problems exceeding 90%. In practice, a deployed SpectraNet must respond to a dynamic dataset, in which the number of observations of known RSOs grow and new RSO classes are periodically added to the problem. We present a paradigm in which model retraining with incremental learning improves deployed effectiveness of SpectraNet and capitalizes on the natural increase in size and diversity of the underlying dataset afforded by autonomous collection. This work provides a pathway for the deployment of SpectraNet-class models to live telescope operations. We demonstrate the dynamics of SpectraNet with the introduction of new RSOs, determine that a minimum of 100 observations is needed to reach performance on a new class of RSO, and show how a subset of incremental learning techniques are well suited for spectroscopic imagery.

1. INTRODUCTION

Effective space traffic management requires positive identification of resident space objects (RSOs). A new technique for learned recognition of satellites using spectroscopic deep networks (SpectraNets) provides simulated and observational proof of concept for positively identifying spatially unresolved targets [6]. The application of SpectraNets to ground based observations of geosynchronous RSOs (see, for example, Figure 1) promises to significantly enrich traditionally position-only space traffic management databases.

But, the path from proof of concept to deployed technology can be treacherous for applications of deep learning. Challenges emerge as techniques like SpectraNet mature. Datasets will grow in depth and breadth: ongoing observations increase the overall number of examples, and expanded target coverage increases the number of classes to be predicted. RSOs will accrue damage and weather with time, introducing class drift. Sensors age and are upgraded or replaced, introducing drift or shock in the translation from photon- to digitally-encoded data. We refer to these types of problems as *deep learning for engineering operational systems*, and note that they lay on the critical pathway to technology transitions.

In this work we investigate the highest impact problem in deploying deep learning algorithms to positive identification: continuous growth of the set of identifiable objects. This occurs rapidly at first, as additional RSOs are observed and codified as classes, and eventually settles to a continuous growth as new RSOs are launched into orbit. A new research effort is not plausible for each new launch–deep learning technologies must be transitioned with the infrastructure needed to expand in scope.

In principle, a model retrained using the full dataset will result in the most performant model. Indeed, deployed technologies should be periodically fully retrained. Between full retraining, transitioned models may efficiently gain

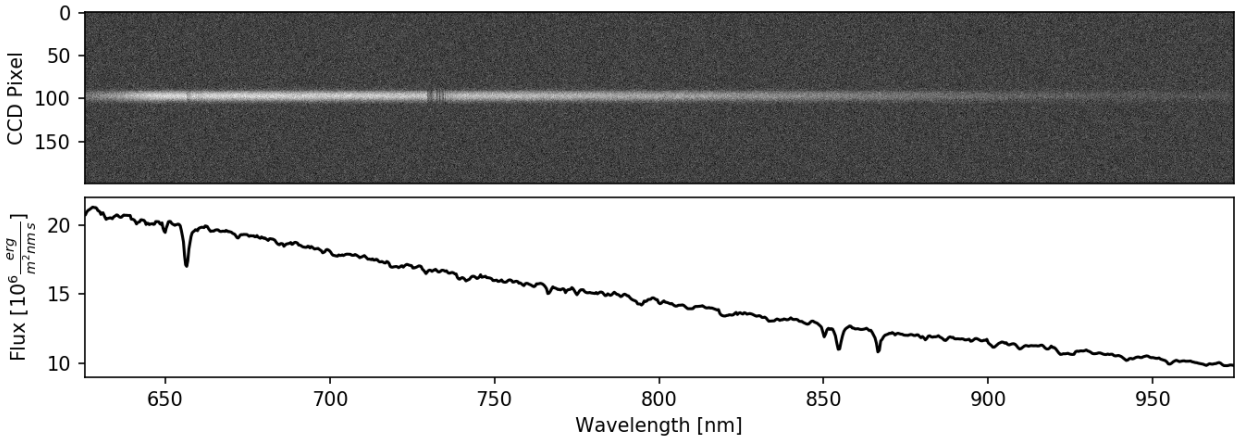

Fig. 1: **Top panel**: A simulated raw FPA observation of 18 Scorpii, a star which is a close analog of our Sun [3], such that this spectrum and FPA frame are typical of resident space objects reflecting solar radiation. These raw frames are used to train models in this paper. **Bottom**: The 1-D reduced spectrum of 18 Scorpii after a raw FPA frame is fully calibrated [1].

in function by incrementally retraining on newly observed data. Incremental training is computationally efficient compared to full retraining, allowing for high frequency model updates on less powerful infrastructure. Eventually, full retraining becomes prohibitive as datasets grow to unmanageable sizes, and incremental learning becomes critical even for major periodic model updates.

Contributions introduced by this work include:

- Demonstrate the application of incremental learning to maximize the effectiveness and stability of deep learning frameworks deployed against problems in space traffic management.
- Perform an ablation study of three incremental learning techniques: bias correction layers, balanced fine tuning, knowledge distillation loss
- Developed retraining infrastructure critical for the deployment of operational deep learning technologies.
- Show that a set of at least 100 observations of a new class should be collected before triggering an incremental learning iteration
- Demonstrate the effectiveness of wide initial convolutional kernels to longslit spectrographic scientific imagery.

In Section 2 we outline prior work in astronomical deep learning and incremental learning. Section 3 describes the details of our formulation. Sections 4 and 5 describe the datasets used for this work and the experiments run on that data. We describe our results in Section 6, and close with concluding remarks in 7.

2. PRIOR WORK

2.1 Deep Learning and Astronomical Imagery

High contrast scientific imagery−such as frames collected by ground based telescopes−offer unique challenges to widely available baseline deep learning datasets. Still, the benefits of learned networks applied to scientific imagery are large in fields like medical imaging and space traffic management. The applicability of deep neural networks to ground based telescope imagery was introduced by [4], who trained object detection algorithms on unresolved imagery. In [5], the authors demonstrate training a binary classifier to detect the presence of closely spaced objects in simulated raw spectroscopic imagery at geostationary orbits. In [6], high accuracy classification of resident space objects based on raw spectroscopic imagery is demonstrated in simulation and using a small set of on-sky data.

Table 1: Simulated Dataset Parameters

Set	Parameter	Value
Atmosphere [10] [9]	Seeing FWHM	U(0.4, 2.0)
	Airmass	U([1.0, 1.5, 2.0, 2.5, 3.0])
	PWV	0.5
	Observatory altitude [m]	3060
Instrument	Grating [grooves/mm, blaze]	150, 800
	Pixel Pitch [μm]	20
	Spectral resolution [nm]	6.5
	Dark Current [e^{-1}/ph/s]	0.005
	Read Noise [e^{-1} RMS]	3
	Bias [ADU/pix]	600
	PSF mode	Gaussian
	Gain	1.0
Exposure	Target N_{ADU} [ph/pixel column]	1000

With the advent of learned solutions to pressing problems in space traffic management, we push towards the development of infrastructure needed to deploy and support deep networks to operational systems.

2.2 Incremental Learning

Applying operational incremental deep learning classifier must train across constantly acquired new data, maintain performance on seen classes while learning to classify new ones, and maintain a constant model backbone [2]. We implement and measure the effectiveness of the following set of techniques against raw longslit spectrsocopy:

Exemplar Memory selects new examples to store for future training and removes other samples [2]. Samples are processed via a herding selection [12], keeping those closest to the mean sample for any given class. The number of samples per class retained ($N_{exemplar}$) is an adjustable hyperparameter, and the technique is applied once per new class addition. In practice, a method of repopulating or updating exemplar memory will be required to handle not only additional classes but re-observation of existing classes.

Distillation Loss retains information from previously fit classes [8]. This loss is applied to classification layers from old class layers while categorical cross-entropy is applied to all classification layers. In our treatment, distillation loss is added to categorical cross-entropy classification loss to form what [2] refer to as "cross-distilled loss".

Balanced Fine Tuning corrects for class imbalances which can arise when the number of samples from old classes ($N_{exemplar}$) and the number of samples for new classes are unbalanced. This is accomplished by a fine tuned training stage with a significantly lower learning rate and classes forced to be in balance by normalizing all class counts to $N_{exemplar}$.

Bias Correction Layers offer an alternate to Balanced Fine Tuning for balancing the bias towards newly introduced classes [13]. This technique adds a two parameter linear model after the final fully connected layer of a model. This final layer is considered the "second stage" of training, and training datasets (exemplars and full samples of new classes) are split internally into a first stage training and second stage validation set. This second stage validation is used to train bias correction in the final Bias correction layers.

3. FORMULATION

We implemented a retraining engine, developed in Python and Tensorflow to modify the training treatment of an existing deep network. In this way, a model trained on a full dataset can be retrained with the efficiency benefits of incremental learning as new data is collected. The updated model weights can be redeployed to the same system utilizing the original model without modification.

We adopt a wide kernel ResNet-152W, which offers advantages for the specifics of our scientific imagery (Fig. 1) [6].

Table 2: Single Replica Broad Hyperparameter Study

Hyperparameter	Values	Comments
Backbone (Batch)	ResNet-152 (10) ResNet-152W (60) AttentionResNet-92 (9)	Batch Size dictated by model size and hardware. ResNet-152W outperformed other backbones [1]
Learning Rate	$10^{-3}, 10^{-4}, 10^{-5}$	
Regularization Kernel + Bias	$10^{-2}, 10^{-3}, 10^{-4}$	
Exemplar Memory $N_{exemplar}$ / class	0, 50, 200	$N_{exemplar}$ = 0 showed catastrophic forgetting
Distillation Loss	Yes, No	
Balanced Fine Tuning	Yes, No	
Bias Correction	Yes, No	

[1] ResNet-152W offers a few advantages conducive to this work (see §6.1)

We provide an experiment showing that wide frames (1340 x 200) capturing low resolution spectra train faster with wider initial convolutional kernels in §6.1.

4. DATASETS

We construct a spectrograph in simulation designed from off the shelf components, enabling realistic simulations of focal plane array (FPA) output. We utilize a proprietary radiometry code and adopt the Cerro Paranal Advanced Sky Model [10, 9] to provide atmospheric transmission and emission based on parameters including precipitable water vapor (PWV), airmass, and observatory altitude. Simulated images are 200x1340x1 in height, width, and channels (Fig 1), and show a characteristic horizontal strip of exposed pixels; this strip is the result of an unresolved point spread function smeared along the horizontal image axis as a function of photon wavelength. In this way, a single channel FPA resolves rich spectral−or color−information. Our simulated instrument design captures a spectral energy distribution between 630 and 980 nanometers.

We generate a large set of simulated spectra of 15 RSOs maintaining nadir orbits in which the spacecraft are pointed directly towards earth's surface. This approximates normal operations for most space assets. Simulation parameters are tabulated in Table 1.

5. EXPERIMENT

We conduct a large, coarse hyperparameter sweep to inform a focused study. Parameter settings for the coarse and focused studies are detailed in Table 2 & Table 3, respectively. We performed one replica each for the coarse study, and three replicas for each hyperparameter combination in the focused study. Performance metrics for the focused study are reported as (max $\pm$ standard deviation) to provide a rough measurement on spread in performance between training runs based on weight initialization.

We report on five "treatments" of incremental learning, using either Bias Correction, Balanced Fine tuning, or no correction and combining those bias treatments with distillation loss or no loss modifications.

5.1 Stability

We perform a training stability study to inform our choice of model backbone and to explore the hypothesis presented in [6] that wide initial convolutional kernels are well suited to raw spectroscopic imagery. We accomplish this by selecting three backbones (AttentionResNet92 [11], ResNet-152 [7] and ResNet-152W [7, 6]), two learning rates (10^{-4}, 10^{-5}), and 10 values of training set size (measured in observations per class: 10, 25, 50, 75, 100, 200, 400, 600, 800, 1000).

Table 3: Three Replica Focused Study

Hyperparameter	Values	Comments
Backbone (Batch)	ResNet-152W (60)	Batch Size dictated by model size and hardware
Learning Rate	$10^{-3}, 10^{-4}$	
Regularization Kernel + Bias	$0, 10^{-3}$	No regularization case added
Exemplar Memory $N_{exemplar}$ / class	50, 200, 300	$N_{exemplar}$ = 300 added, 0 removed
Distillation Loss	Yes, No	
Balanced Fine Tuning	Yes, No	
Bias Correction	Yes, No	

For each of these parameter combinations we train 30 replicas, each with early stopping patience of 20 to allow models which have stabilized in validation accuracy to stop training. Each replica is limited to a maximum training time of four hours.

5.2 Baseline

For each incremental learning experiment we train baseline models on the full 15 class, 500 examples per class dataset matching backbone, learning rate, and regularization hyperparameters for comparison.

5.3 Incremental Learning

We begin by training a model on 2 classes for 100 epochs, then extending the experiment by 1 class and 100 epochs until all 15 classes have been added to the experiment. After each 100 epoch training cycle, we store $N_{exemplar}$ examples from the added class(es) to the exemplar memory bank,

6. RESULTS

Table 4 contains condensed results for the 288 training runs completed in our fine study (outlined in §5, Tbl 3). Table 4 is organized by descending treatment performance. The first two columns provide treatment and best accuracy for that treatment, and the next four columns tabulate the five best hyperparameter combinations (and performance) for that treatment. We discuss each treatment in the sections that follow.

6.1 Training Stability

We quantify the qualitative finding of enhanced SpectraNet performance when widening the initial convolutional layer of ResNet−kernel of 7x49 instead of 7x7 and stride of 2x12 instead of 2x2. [6].

In Figure 2 we visualize the effects of training examples per class on classification accuracy. For training sets containing 50 examples per class or fewer, performance is negligible. Once example counts pass 75, residual networks with faster learning rates begin to perform, although that performance is highly stochastic with respect to weight initialization. Once the dataset grows to 100 or more examples per class, ResNet-152W with faster learning rate performs exceptionally well in all replicas, significantly outperforming all other baselines. ResNet-152 with higher learning rate performs well at over 200 examples per class, and AttentionResNet-92 and lower learning rate models lag behind.

With our training time limit of four hours per replica, these results are confounded by situations in which models don't have enough time to train to their maximum performance. We resolve this confusion in Figure 3, where we visualize the training time, in minutes, used for these experiments. These plots have two main modes. First, when box plots rest at 240 minutes the models would likely continue to improve in performance given more training time. In other cases, the models stabilize in performance and stop training due to an implementation of early stopping patience.

Combining the observations of Figs 2 & 3, we can intuit that the performance gains of ResNet-152W are largely efficiency based. With a smaller memory footprint and resulting larger batch size, training runs are able to reach higher performance more rapidly.

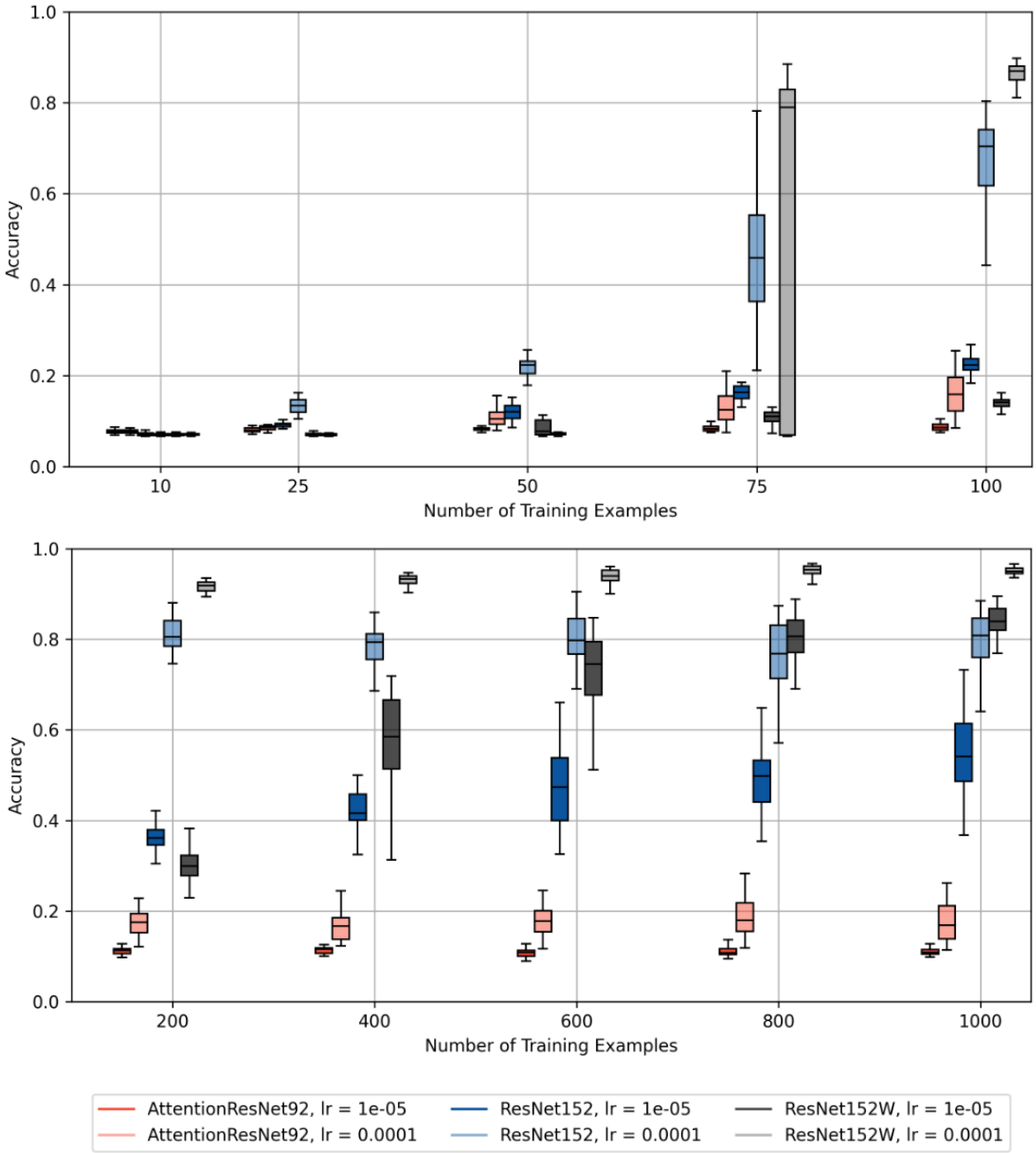

Fig. 2: Training accuracy of SpectraNet against dataset size and backbone. **Top panel**: 10 to 100 training examples per class, with AttentionResNet92 in shades of red (left two boxes per grouping), ResNet-152 in blues (center two boxes) and ResNet-152W in grays (right two boxes). **Bottom**: 200 - 1000 training examples per class.

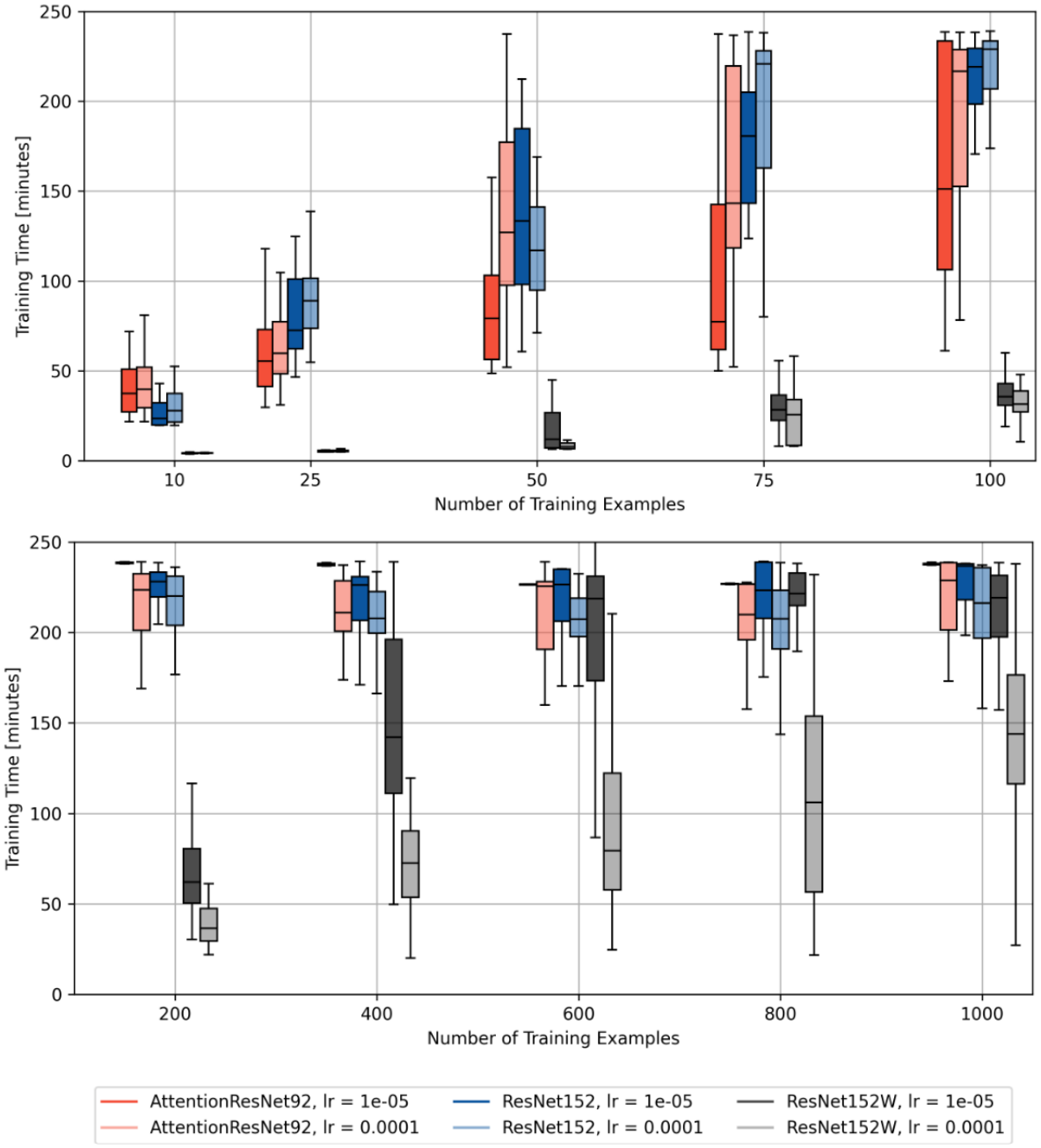

Fig. 3: Training time of SpectraNet against dataset size and backbone. Models were trained with early stopping patience and a maximum time of 240 minutes. **Top panel**: 10 to 100 training examples per class, with AttentionResNet92 in shades of red (left two boxes per grouping), ResNet-152 in blues (center two boxes) and ResNet-152W in grays (right two boxes). **Bottom**: 200 - 1000 training examples per class.

Treatment	Best Accuracy	Top 5 Param. Sets				
		Accuracy	Perf. Loss	LR	Reg	$N_{exemplar}$
BFT	93.73 ± 0.75	93.73 ± 0.75	2.93	0.0001	0	300
		93.33 ± 1.65	2.13	0.0001	0.001	300
		92.87 ± 2.05	3.47	0.0010	0	300
		92.80 ± 0.58	2.07	0.0010	0.001	300
		91.07 ± 0.37	4.40	0.0001	0.001	200
BiC	93.67 ± 0.62	93.67 ± 0.62	1.80	0.0001	0.001	300
		93.20 ± 1.15	1.67	0.0010	0.001	300
		93.13 ± 0.16	3.53	0.0001	0	300
		91.73 ± 6.94	4.60	0.0010	0	200
		91.40 ± 0.69	5.27	0.0001	0	200
Dist. Loss + BFT	82.53 ± 2.59	82.53 ± 2.59	14.13	0.0001	0	300
		82.40 ± 2.16	13.07	0.0001	0.001	300
		82.00 ± 3.48	12.87	0.0010	0.001	300
		78.93 ± 1.52	17.40	0.0010	0	300
		77.00 ± 9.85	17.87	0.0010	0.001	200
Dist. Loss	80.07 ± 4.91	80.07 ± 4.91	16.27	0.0010	0	300
		78.73 ± 1.98	17.93	0.0001	0	300
		78.00 ± 1.23	17.47	0.0001	0.001	300
		74.73 ± 1.72	20.13	0.0010	0.001	300
		65.27 ± 7.00	29.60	0.0010	0.001	200
Dist. Loss + BiC	78.29 ± 1.84	78.29 ± 1.84	17.18	0.0001	0.001	300
		75.86 ± 1.49	20.48	0.0010	0	300
		75.13 ± 2.86	19.73	0.0010	0.001	300
		74.53 ± 2.53	22.13	0.0001	0	300
		62.87 ± 4.90	33.80	0.0001	0	200

Table 4: Condensed results from a hyperparameter study showing that balanced fine tuning (BFT) and bias correction layers (BiC) perform well in our domain while distillation loss does not. Treatments, presented in descending best accuracy, are tabulated along the top five best performing hyperparameter sets (and resulting loss in performance over that hyperparameter baseline). We discuss these results in depth in §6.

6.2 Bias Correction and Balanced Fine tuning

Our implementation of Bias Correction Layers introduced by [13], and Balanced Fine Tuning of [2] perform equally well when applied without Distillation loss. With model classification accuracies of ∼93% and performance losses of only a few percent (∼3%), these two treatments offer a solution to incrementally trained spectroscopic Positive ID models deployed to operational systems.

6.3 Distillation Loss

The poor performance of distillation loss aligns with the results of [2], where it is hypothesized as a combination of weak representation of old classes and a lack of exemplar memory. In the course hyperparameter study, we found catastrophic forgetting in experiments with distillation loss and no exemplar memory, and weak performance when combined with exemplar memory.

We dropped $N_{exemplar} = 0$ for our fine study, but still find that incremental learning treatments including distillation loss perform significantly worse (∼80% accuracy and ∼ 15% performance loss) compared to treatments without distillation loss (∼93% accuracy and ∼3% performance loss)

6.4 Exemplar Memory

We note that top performing models (those above 90% classifier accuracy) all required $N_{exemplar} = 200$ or 300, and with our baseline dataset containing 500 examples per class (meaning that $N_{exemplar} = 500$ would effectively represent the baseline training case), we note that this is a 15 class problem and the performance gains of incremental learning are still significant.

The baseline model is trained on the full dataset of 7500 examples, while the incrementally learned model is trained on 2 classes for 100 epochs (1000 examples), and then 3 classes (900-1100 examples: 400 or 600 $N_{exemplar}$ and 500 examples from the new class), and so on. In total, after the 1400 training epochs needed to train an incrementally learned model, that model will have seen 3.18 to 4.37 million examples, while the baseline model will have seen 11.3 million.

Furthermore, if a high performing baseline model were deployed on sky and then fully retrained to include a 16th class, that model would need to train over 11.3 million examples again, while an incremental training run would need to see 0.37 to 0.53 million examples, requiring significantly less time or significantly less compute at the cost of 2-3% performance loss.

7. CONCLUSIONS

In this work we implement techniques from the field of incremental learning to explore the technologies needed for the transition of deep learning solutions to operational systems. We conduct an experiment to determine that at least 100 examples of a new RSO should be collected before hoping to successfully classify new observations of that RSO. Finally, we present an experiment demonstrating that the wide scientific imagery characteristic of longslit spectroscopy responds well to the ResNet-152 backbone with widened initial kernels.

We demonstrate how incremental model training employing either balanced fine tuning or bias correction layers offers a pathway to continuous gain of function for deployed models, allowing rapid retraining and classification on new classes. This, combined with periodic rebaselining on the full dataset on dedicated training compute resolves one of the issues of deep learning for engineering operational systems in the field of space domain awareness.

[1] Ralph C Bohlin, Karl D Gordon, and P.-E Tremblay. Dissertation Summary Techniques and Review of Absolute Flux Calibration from the Ultraviolet to the Mid-Infrared. Technical report, 2014.

[2] Francisco M. Castro, Manuel J. Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-End Incremental Learning. 7 2018.

[3] G. Cayrel de Strobel. Stars resembling the Sun. *Astronomy and Astrophysics Review*, 7(3), 10 1996.

[4] Justin Fletcher, Ian Mcquaid, Peter Thomas, Jeremiah Sanders, and Greg Martin. Feature-Based Satellite Detection using Convolutional Neural Networks. *AMOS*, 2019.

[5] J. Zachary Gazak, Justin Fletcher, Ryan Swindle, and Ian Mcquaid. Exploiting Spatial Information in Raw Spectroscopic Imagery using Convolutional Neural. *AMOS*, 2020.

[6] J. Zachary Gazak, Ian McQuaid, Justin Fletcher, Thomas Swindle, and Matthew Phelps. SpectraNet: Learned Recognition of Artificial Satellites from High Contrast Spectroscopic Imagery. *WACV*, Submitted, 2022.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. 12 2015.

[8] Geoffrey Hinton and Jeff Dean. Distilling the Knowledge in a Neural Network. *NIPS*, 2015.

[9] A. Jones, S. Noll, W. Kausch, C. Szyszka, and S. Kimeswenger. An advanced scattered moonlight model for Cerro Paranal. *Astronomy and Astrophysics*, 560, 12 2013.

[10] S. Noll, W. Kausch, M. Barden, A. M. Jones, C. Szyszka, S. Kimeswenger, and J. Vinther. An atmospheric radiation model for Cerro Paranal: I. the optical spectral range. *Astronomy and Astrophysics*, 543, 2012.

[11] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual Attention Network for Image Classification. 4 2017.

[12] Max Welling and Donald Bren. Herding Dynamical Weights to Learn. *ICML*, 2009.

[13] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large Scale Incremental Learning. 5 2019.