Convolutional neural networks and satellite imagery: How deep is necessary?

Anonymous Author(s)
Affiliation
Address
email

Abstract

Applying off-the-shelf models (e.g., ResNet) to satellite imagery has become standard practice. While convolutional neural networks (CNNs) have been shown to outperform baseline methods in remote sensing prediction tasks, differences in satellite and natural images (i.e., images that comprise common datasets like ImageNet and CIFAR-10) may make ResNet-type models overkill for many satellite imagery tasks. In this paper, we present a comparison of off-the-shelf CNNs to a much smaller CNN over a range of satellite imagery tasks and show that a CNN with significantly fewer parameters performs on par with standard CNN architectures for five out of six tasks. Our findings are especially pertinent to those working with satellite imagery who face computational constraints.

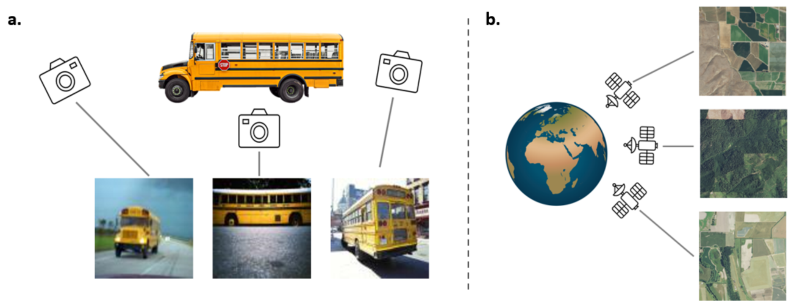
1 Introduction

Machine learning has continually proven successful in informing sustainability-related tasks from satellite imagery. Example tasks include crop type mapping (1; 2), poverty prediction (3), and water quality monitoring (4; 5). It has become standard practice to apply off-the-shelf models (e.g., ResNet) to satellite imagery. While CNNs have been shown to outperform baseline methods in remote sensing tasks, there are substantial enough differences between natural images (i.e., images that comprise common datasets like ImageNet and CIFAR-10) and satellite images, that off-the-shelf CNNs may be unnecessarily large for many satellite imagery analyses.

Prior to deep learning, handcrafted features were common in satellite imagery analysis. Features generally consisted of low-level color and texture descriptors, such as color histograms (6). Recently, basic color descriptors have been shown to be highly effective in discriminating coffee/non-coffee scenes (6) and simple statistics over images have been shown to be informative in predicting bird distributions from satellite imagery (7). As filters in earlier layers of CNNs generally pick up on color and texture and the later layers are more representative of concepts (8), many satellite imagery tasks may not require significantly deep networks.

Parameter reduction in neural networks is a popular area of research (9; 10; 11; 12; 13). Not only are smaller models beneficial from a training perspective (e.g., time and carbon footprint), but in many cases smaller models are necessary when deployed on devices with limited computational resources. Methods for designing smaller neural networks generally involve compressing pretrained networks or designing smaller networks (10). MobileNet (10; 14) and SqueezeNet (9) are two common architectures optimized for fewer parameters but capable of achieving ResNet-level accuracy. While both of these networks are indeed smaller than ResNet18, they still contain roughly a million parameters. We hypothesize that common CNN architectures, even those optimized to have fewer parameters than ResNet, are overkill for many satellite imagery tasks.

Submitted to Computational Sustainability Workshop at NeurIPS 2023. Do not distribute.



	Natural imagery	Satellite imagery
Camera angle	varied	fixed
Camera proximity	varied	fixed
Object occlusion	yes	N/A
Object orientation	generally fixed	N/A

Figure 1: Natural versus satellite images. Left: three school buses captured at different viewing angles and proximities (15). There is no occlusion in the school bus images and the orientation of all school buses is the same (wheels on the ground, below the body of the buses). Right: three satellite images with the same camera angle, proximity, and orientation as images from the same satellite are captured at a fixed, bird's-eye view.

We performed an analysis on CNN architectures to determine if larger models, such as ResNet and ResNet alternatives, are necessary for satellite imagery tasks. Specifically, we compared several off-the-shelf models to a simple, shallow CNN on multiple regression and classification tasks. Our main contributions are: 1) an analysis of modern CNN architectures across several satellite imagery tasks, and 2) results showing that a shallow CNN, with millions of fewer parameters than ResNet18, is comparable to standard ResNet-type models for five out of six satellite imagery tasks.

2 Natural versus satellite images

Satellite images differ from natural images in several key ways. Most notably, satellite images are captured in a significantly more structured manner compared to natural images. There are several degrees of freedom when capturing natural images: 1) camera proximity from the subject, 2) camera angle and 3) camera orientation relative to the subject, and 4) items occluding the subject (Figure 1a). Satellite images within publicly available datasets, such as Landsat and Sentinel, are captured at a relatively fixed distance and orientation to Earth and rarely suffer from occlusion (Figure 1b). This fixed nature in which satellite images are captured enforces that images taken from the same satellite have essentially the same scale. It is possible for atmospheric conditions (e.g., clouds) to obscure satellite images; however, a common preprocessing step is to filter out days/images that contain heavy atmospheric conditions. The fewer degrees of freedom in which satellite images are captured likely simplifies the complexity of many prediction tasks.

CNNs for natural images must have incredibly large capacities (i.e., millions of parameters) to represent classes well. Accounting for different camera viewing angles and distances, and the potential of object occlusion makes object classification a difficult task. CNNs must learn many representations of the same class (e.g., a school bus is a school bus no matter what angle or position the image is captured from and whether or not the bus is partially occluded - Figure 1a). Apart from the differences in how natural and satellite images are captured, there is evidence that color and texture descriptors are effective features in satellite imagery analysis (6). Such features are detected in early CNN layers, meaning relatively deep CNNs are potentially unnecessary. Below, we test if these differences between domains does indeed simplify satellite imagery analysis and determine if shallower CNN architectures are capable of performing as well as larger, off-the-shelf models.

Table 1: **Summary of architectures.** The number of parameters and parameter reductions compared to ResNet18 and memory required to train each architecture.

Architecture	# parameters	Reduction from ResNet18	Memory
ResNet18	11.2M	-	42.80MB
MobileNetV3	1.7M	7x	6.34MB
SqueezeNet	0.7M	16x	2.76 MB
ShallowCNN	0.1M	87x	0.49MB

3 Experiments

We compared four CNN architectures on three regression and three classification tasks. We compared a basic, shallow CNN, hereafter *ShallowCNN*, to three pretrained off-the-shelf CNNs: ResNet18, MobileNetV3-small, and SqueezeNet. Our ShallowCNN has a straightforward (conv2d-batch norm-relu-pooling) architecture (Table A1). For each of the tasks and models, with the exception of Brazilian Coffee Scenes, we used 10-fold cross-validation to train 10 models in order to quantify variation in model performance. We used 5-fold cross-validation on Brazilian Coffee Scenes as the dataset came with 5 predefined folds. We fine-tuned all layers of the pretrained models and trained ShallowCNN from scratch, for each task respectively. We evaluated the regression tasks with R^2, mean squared error (MSE), and mean absolute error (MAE) and evaluated the classification tasks with overall accuracy, precision, and recall. All three classification tasks have roughly equal class balance, therefore, we report the average overall accuracy, precision, and recall across classes. In addition to comparing the CNN architectures across tasks, we evaluated the impact of dataset size on model performance. Details on data preprocessing and model training are in the appendix.

We compared the four CNN architectures by modeling six satellite imagery tasks. In order to generalize across tasks, we selected a set of problems that range in difficulty and dataset size. We modeled three regression tasks from Rolf et al. (2021): percent forest cover, nighttime light intensity, and elevation (Table A2, Figure A1) Additionally, we modeled three classification tasks: crop type, Brazilian Coffee Scenes (6), and UCMerced Land-use (17) (Table A2, Figure A1). We selected the Brazilian Coffee Scenes and UCMerced Land-use datasets as they have been commonly used in previous remote sensing studies (18; 19; 20) and created our own crop type dataset. A detailed description of the tasks can be found in the appendix.

4 Results and Discussion

ShallowCNN, with 87 times fewer parameters than ResNet18 (Table 1), was within two standard deviations of ResNet18 in three of the six tasks and exceeded ResNet18 by over four standard deviations in crop type mapping (Table 2). In addition to crop type mapping, ShallowCNN also achieved the highest accuracy for coffee scene identification (Table 2b). The only case in which ShallowCNN had significantly degraded performance is in the land use problem (Table 2b). In comparing the images from all tasks (Figure A1), the land use images are visually the most similar to ImageNet, while the other tasks are likely more reliant on color and texture descriptors. The land use task may require higher-level features to distinguish classes. As the deeper layers of CNNs are generally more representative of concepts (8), satellite imagery tasks which are visually similar to ImageNet may benefit from larger CNNs. Further work is needed to understand which satelliteimagery tasks are better suited for ShallowCNN versus ResNet18-type models.

There are computational benefits to ShallowCNN. While ShallowCNN takes almost as long to train as it does to fine-tune the larger pretrained models, ShallowCNN requires significantly less memory to train (Table 1). If computational resources are limited, training multiple larger models at the same time may not be feasible, however, it may be possible to train several ShallowCNNs concurrently. In addition to benefits in training, ShallowCNN's smaller model size is beneficial if models are being deployed externally and space is limited by hardware (e.g., analyzing images real time on a drone or satellite). Further work should investigate the potential of a pretrained ShallowCNN for satellite imagery.

Table 2: **Summary of model performances across tasks**. Regression tasks are evaluated on MSE and classification tasks on accuracy (other metrics showed similar trends). The reported accuracy is averaged across all classes since all classification tasks have good class balance. Reported results are averaged across cross-validation folds (10 folds for all but Coffee, which had 5), plus or minus their standard deviation. Models are ordered from greatest number of parameters (top) to fewest (bottom).

a. Regression tasks - MSE (x10^{-3})

Model	Percent forest cover	Nighttime light intensity	Elevation
ResNet18	**6.06 ± 0.31**	3.40 ± 0.14	7.55 ± 2.52
MobileNetV3	7.20 ± 1.12	**3.22 ± 0.26**	**7.00 ± 2.32**
SqueezeNet	7.92 ± 1.36	3.54 ± 0.23	8.40 ± 2.63
ShallowCNN	6.85 ± 2.33	3.60 ± 0.26	10.80 ± 2.65

b. Classification tasks - overall accuracy (%)

Model	Crop type	Land use	Coffee
ResNet18	93.35 ± 0.38	**98.89 ± 0.47**	91.28 ± 2.19
MobileNetV3	93.75 ± 0.45	96.53 ± 2.86	88.33 ± 1.48
SqueezeNet	92.80 ± 0.34	91.00 ± 1.90	90.23 ± 0.93
ShallowCNN	**95.04 ± 0.33**	89.00 ± 1.87	**92.13 ± 1.02**

A significant benefit to ShallowCNN is its performance on small datasets, an issue common among sustainability-related tasks. Although a common principle in machine learning says that smaller models should be favored for simpler prediction problems, it is still common practice to apply ResNet-type models to satellite imagery tasks. Until the training size is reduced to 8k images, there is a small, roughly linear increase in MSE across all models (Figure 2). Once the training set size falls below 8k, there is an exponential increase in MSE for ResNet and SqueezeNet and a much smaller increase in MSE for ShallowCNN and MobileNet (Figure 2). While MobileNet outperforms ShallowCNN on smaller datasets, ShallowCNN's smaller memory requirements may outweigh the difference in performance for some applications. Further work should investigate why MobileNetV3, which has significantly more parameters than ShallowCNN and SqueezeNet, outperforms both methods in the small data regime.

Figure 2: MSE for nighttime light intensity as dataset size is reduced. Note that the x-axis is decreasing.

5 Conclusion

Evaluating the performance of smaller, non-standard deep architectures is generally underexplored (21). This is especially of interest when applying CNNs to new domains where there are fundamental differences between the domain images and images that comprise common computer vision datasets (i.e., natural images). We compared a small three layer CNN, ShallowCNN, to much larger off-the-shelf architectures and showed that in most cases ShallowCNN has comparable performance to ResNet-type models. Our results align with other studies which have shown that simple CNN architectures perform well on satellite imagery tasks (22; 23). Our study differs from the previous studies in that it is the first to baseline the performance of modern pretrained architectures with a smaller CNN across several remote sensing tasks, whereas the previous studies investigated the viability of CNNs compared to other deep learning methods. Our results are of special concern to those who have computational constraints, whether in model training or in model deployment.

References

[1] Sherrie Wang, Stefania Di Tommaso, Joey Faulkner, Thomas Friedel, Alexander Kennepohl, Rob Strey, and David B Lobell. Mapping crop types in southeast india with smartphone crowdsourcing and deep learning. *Remote Sensing*, 12(18):2957, 2020.

[2] Gabriel Tseng, Hannah Kerner, Catherine Nakalembe, and Inbal Becker-Reshef. Learning to predict crop type from heterogeneous sparse labels using meta-learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1111–1120, June 2021.

[3] Neal Jean, Marshall Burke, Michael Xie, W. Matthew Davis, David B. Lobell, and Stefano Ermon. Combining satellite imagery and machine learning to predict poverty. *Science*, 353 (6301):790–794, 2016. doi: 10.1126/science.aaf7894. URL `https://www.science.org/doi/abs/10.1126/science.aaf7894`.

[4] Nima Pahlevan, Brandon Smith, John Schalles, Caren Binding, Zhigang Cao, Ronghua Ma, Krista Alikas, Kersti Kangro, Daniela Gurlin, Nguyn Hà, et al. Seamless retrievals of chlorophyll-a from sentinel-2 (msi) and sentinel-3 (olci) in inland and coastal waters: A machine-learning approach. *Remote Sensing of Environment*, 240:111604, 2020.

[5] Jeremy Kravitz, Mark Matthews, Lisl Lain, Sarah Fawcett, and Stewart Bernard. Potential for high fidelity global mapping of common inland water quality products at high spatial and temporal resolutions based on a synthetic data and machine learning approach. *Frontiers in Environmental Science*, 9, 2021. ISSN 2296-665X. doi: 10.3389/fenvs.2021.587660. URL `https://www.frontiersin.org/articles/10.3389/fenvs.2021.587660`.

[6] Otávio AB Penatti, Keiller Nogueira, and Jefersson A Dos Santos. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 44–51, 2015.

[7] Laurel M Hopkins, Tyler A Hallman, John Kilbride, W Douglas Robinson, and Rebecca A Hutchinson. A comparison of remotely sensed environmental predictors for avian distributions. *Landscape Ecology*, 37(4):997–1016, 2022.

[8] Bolei Zhou, David Bau, Aude Oliva, and Antonio Torralba. Interpreting deep visual representations via network dissection. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2131–2145, 2018.

[9] Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and lt;0.5mb model size, 2016. URL `https://arxiv.org/abs/1602.07360`.

[10] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[11] Gao Huang, Shichen Liu, Laurens Van der Maaten, and Kilian Q Weinberger. Condensenet: An efficient densenet using learned group convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2752–2761, 2018.

[12] Bichen Wu, Alvin Wan, Xiangyu Yue, Peter Jin, Sicheng Zhao, Noah Golmant, Amir Gholaminejad, Joseph Gonzalez, and Kurt Keutzer. Shift: A zero flop, zero parameter alternative to spatial convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9127–9135, 2018.

[13] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6848–6856, 2018.

[14] A. Howard, M. Sandler, B. Chen, W. Wang, L. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le. Searching for mobilenetv3. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1314–1324, Los Alamitos, CA, USA, nov 2019. IEEE Computer Society. doi: 10.1109/ICCV.2019.00140. URL `https://doi.ieeecomputersociety.org/10.1109/ICCV.2019.00140`.

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. IEEE, 2009.

[16] E. Rolf, J. Proctor, T. Carleton, I. Bolliger, V. Shankar, M. Ishihara, B. Recht, and S. Hsiang. A generalizable and accessible approach to machine learning with global satellite imagery. *Nature Commun*, 12(4392), 2021. doi: 10.1038/s41467-021-24638-z.

[17] Yi Yang and Shawn Newsam. Bag-of-visual-words and spatial extensions for land-use classification. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '10, page 270–279, New York, NY, USA, 2010. Association for Computing Machinery. ISBN 9781450304283. doi: 10.1145/1869790.1869829. URL `https://doi.org/10.1145/1869790.1869829`.

[18] Marco Castelluccio, Giovanni Poggi, Carlo Sansone, and Luisa Verdoliva. Land use classification in remote sensing images by convolutional neural networks, 2015. URL `https://arxiv.org/abs/1508.00092`.

[19] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017.

[20] Keiller Nogueira, Otávio AB Penatti, and Jefersson A Dos Santos. Towards better exploiting convolutional neural networks for remote sensing scene classification. *Pattern Recognition*, 61: 539–556, 2017.

[21] Maithra Raghu, Chiyuan Zhang, Jon Kleinberg, and Samy Bengio. Transfusion: Understanding transfer learning for medical imaging. *Advances in neural information processing systems*, 32, 2019.

[22] Saikat Basu, Sangram Ganguly, Supratik Mukhopadhyay, Robert DiBiano, Manohar Karki, and Ramakrishna R. Nemani. Deepsat - A learning framework for satellite imagery. *CoRR*, abs/1509.03602, 2015. URL `http://arxiv.org/abs/1509.03602`.

[23] Yanfei Zhong, Feng Fei, Yanfei Liu, Bei Zhao, Hongzan Jiao, and Liangpei Zhang. Satcnn: satellite image dataset classification using agile convolutional neural networks. *Remote Sensing Letters*, 8(2):136–145, 2017. doi: 10.1080/2150704X.2016.1235299. URL `https://doi.org/10.1080/2150704X.2016.1235299`.

[24] National Geospatial Data Asset (NGDA) NAIP Imagery. `http://gis.apfo.usda.gov/arcgis/rest/services/NAIP`, March 2015.

[25] Collin Homer, Jon Dewitz, Limin Yang, Suming Jin, Patrick Danielson, George Xian, John Coulston, Nathaniel Herold, James Wickham, and Kevin Megown. Completion of the 2011 national land cover database for the conterminous united states-representing a decade of land cover change information. *Photogrammetric Engineering and Remote Sensing*, 81(5):345–354, May 2015.

[26] Roozbeh Valavi, Jane Elith, José J. Lahoz-Monfort, and Gurutzeta Guillera-Arroita. blockcv: An r package for generating spatially or environmentally separated folds for k-fold cross-validation of species distribution models. *Methods in Ecology and Evolution*, 10(2):225–232, 2019. doi: 10.1111/2041-210X.13107.

Appendix

A1 ShallowCNN

Our shallowCNN is comprised of three convolutional layers. The weights in conv1 are taken from the first layer of ResNet18 pretrained on ImageNet as using pretrained weights in the initial layer can lead to faster convergence (21). To match ResNet18, the first convolutional layer has 7x7 filters. The remaining two layers have 3x3 filters. See Table A1 for the full architecture.

Table A1: **ShallowCNN architecture.** For the last layer we used a multiclass softmax activation for the classification tasks and a linear activation for the regression tasks.

Layer name	
conv1	7x7, 64 conv; batch norm; ReLU; max pooling
conv2	3x3, 64 conv; batch norm; ReLU; max pooling
conv3	3x3, 128 conv; batch norm; ReLU; max pooling
avg_pool	average pooling
fc	128x64, ReLU, 64xnum_classes
activation	classification: softmax; regression: linear

A2 Tasks

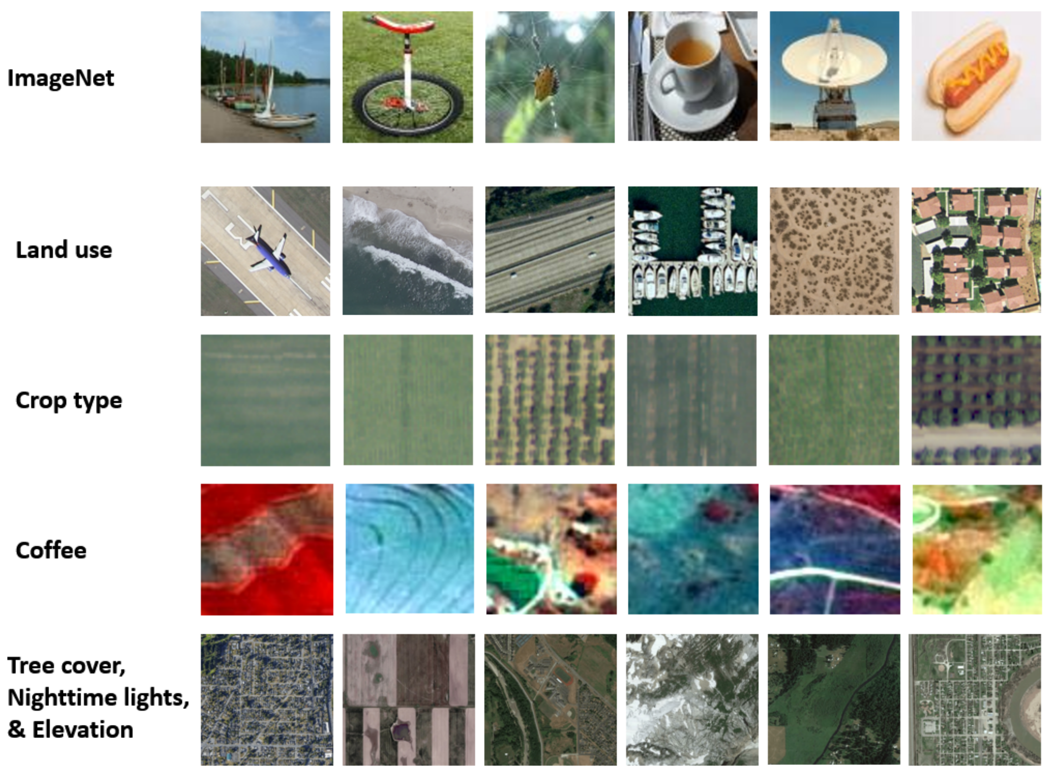


Figure A1: Sample images from ImageNet (15), UC Merced Land-use (17), crop type, Brazilian Coffee Scenes (6) and percent forest/nighttime light intensity/elevation tasks (16). Color and textural information appear more indicative of class in the satellite images as opposed to those of ImageNet.

A2.1 Percent forest cover, nighttime light intensity, and elevation

We purposefully selected a set of satellite imagery tasks that range in difficulty and dataset size. Table A2 outlines the tasks and dataset sizes. We selected three regression tasks from Rolf et al. (2021) with increasing complexity: percent forest cover, nighttime light intensity, and elevation. Forest cover is directly observable from satellite imagery and, therefore, should be the most straightforward to predict. Nighttime light intensity itself is not observable from daytime satellite images, however, proxies for nighttime light intensity (e.g., dense urban areas) are observable. Elevation on the other hand, is much more difficult to estimate solely from a satellite image. Many images may have similar appearances but dramatically different elevations. Images for all three regression tasks were collected from the contiguous United States based on the sampling schemes of Rolf et. al (2021). See Figure A1 for sample images.

A2.2 Crop type

We developed our own crop type dataset from images collected from three regions within the Central Valley of California. We collected National Agricultural Imagery Program (NAIP) imagery (24) from 2012 and derived labels from the National Land Cover Database (NLCD) (25) for the same year. We subset the data to only include the three most commonly occurring crops: tomatoes, almonds, and alfalfa. When assigning labels, we only included images which contained more than 60% of the majority label in the image. In total the dataset consists of 36,000 images with a roughly even split across the three classes.

A2.3 Brazilian Coffee Scenes

SPOT satellite images were collected in 2005 over four counties in Brazil. Images were labeled by agricultural experts and labeled coffee if more than 85% of the pixels contained coffee and non-coffee if less than 10% of the pixels contained coffee. The dataset consists of 2876 images with an equal split of coffee and non-coffee (6).

A2.4 UC Merced Land-use

The UC Merced Land-use dataset consists of aerial images from 21 different land use classes. The classes span categories such as beach, parking lot, buildings, forest, and overpass. The images were collected from 20 cities across the United States. The images were manually annotated and each class contains 100 images (17).

Table A2: **Description of tasks.**

Prediction task	Type	Classes	Dataset size	Image size	Spatial res.
Percent forest cover (16)	regression	1	100k	256x256	~ 4 m
Nighttime light intensity (16)	regression	1	100k	256x256	~ 4 m
Elevation (16)	regression	1	100k	256x256	~ 4 m
Crop type	classification	3	36k	48x48	1 m
Brazilian Coffee Scenes (6)	classification	2	2876	64x64	-
UCMerced Land-use (17)	classification	21	2100	256x256	0.3 m

A3 Data pre-processing

Spatial autocorrelation is a common issue in spatial data and can be problematic as it can artificially overestimate the predictive power of models by having highly correlated datapoints (i.e., datapoints close in geographical space) in both the training and testing sets. To help address spatial autocorrelation, we used the blockCV R package (26) to split datasets with geographic location information into spatial blocks. We then used the spatial blocks to assign data points into 10 folds for cross validation. For the tasks without location information, we randomly split the data into 10 cross validation folds. All three regression tasks and the crop type task have location information (i.e., they were split spatially). We randomly assigned splits for the land use task and used the predefined splits for the coffee dataset.

For image preprocessing we scaled pixel values to be in the range of [0, 1] and subtracted the channel means. During training, we augmented images by performing random horizontal and vertical flips and random rotations in increments of 90°.

A4 Training

We fine-tuned the three off-the-shelf models (ResNet18, MobileNetV3-small, and SqueezeNet) and trained ShallowCNN from scratch. For the three off-the-shelf architectures, we used pretrained weights and updated the fully-connected layer to match the number of outputs for the given task. For all models and all tasks, we performed hyperparameter tuning on the learning rate, weight decay, and batch size. Prior to comparing ShallowCNN to the other models, we experimented with different numbers of convolutional layers and different numbers of convolutions per layer. Across tasks, we found a three layer model with 64, 64, and 128 convolutions to perform the best. We transferred pretrained weights from the first convolutional layer of ResNet18 for the first layer of ShallowCNN as using pretrained weights in the initial layer can lead to faster convergence (21). To match ResNet18, the first convolutional layer has 7x7 filters. The remaining two layers have 3x3 filters and the weights were randomly initialized. See Table A1 for the full architecture.