Efficient Convolutional Neural Networks for Multi-Spectral Image Classification

Jacob J. Senecal
Gianforte School of Computing
Montana State University
Bozeman, USA
jacob.senecal@student.montana.edu

John W. Sheppard
Gianforte School of Computing
Montana State University
Bozeman, USA
john.sheppard@montana.edu

Joseph A. Shaw
Dept. Elec & Computer Engineering
Montana State University
Bozeman, USA
joseph.shaw@montana.edu

***Abstract*—While a great deal of research has been directed towards developing neural network architectures for RGB images, there is a relative dearth of research directed towards developing neural network architectures specifically for multi-spectral and hyper-spectral imagery. We have adapted recent developments in small efficient convolutional neural networks (CNNs), to create a small CNN architecture capable of being trained from scratch to classify 10 band multi-spectral images, using much fewer parameters than popular deep architectures, such as the ResNet or DenseNet architectures. We show that this network provides higher classification accuracy and greater sample efficiency than the same network using RGB images. Further, using a Bayesian version of our CNN architecture we show that a network that is capable of working with multi-spectral imagery significantly reduces the uncertainty associated with class predictions compared to using RGB images.**

***Index Terms*—multi-spectral, convolution, bayesian, uncertainty**

I. Introduction

Sources of hyper-spectral and multi-spectral imagery are becoming increasingly prevalent. For example, the European Space Agency launched the Sentinel-2A satellite in June of 2015, and the Sentinel-2B satellite was launched in March of 2017. Both satellites are capable of imaging in 13 different spectral bands, ranging from 443 nm to 2190 nm. These satellites have a high revisit rate, with every area of Earth beneath these satellites orbital path being imaged every five days. The full Sentinel constellation is expected to generate approximately 8 terabytes of data every day in operation [1]. This type of data can be incredibly valuable in land management, agriculture and forestry, disaster control, humanitarian relief operations, and environmental monitoring [2].

Processing and interpreting this amount of data is a challenge. Machine learning and computer vision techniques could automate the process of extracting useful information from the raw multi-spectral imagery provided by sources such as the Sentinel satellite constellation, and NASA's LANDSAT mission.

However, while the use of machine learning to process and classify generic RGB images has received a great deal of attention, and in general has been quite successful, much less energy has been devoted to extending machine learning models like convolutional neural networks to process the type of multi-spectral imagery provided by space based systems. One issue has been the lack of large, labeled training datasets for satellite imagery, compared to RGB images where there exist several large publicly available datasets such as ImageNet [3] and CIFAR-10 [4]. It is difficult to train a deep convolutional neural network (CNN) on a small dataset without overfitting to the training data due to the CNN's large number of parameters. The increased number of spectral channels in a multi-spectral image compared to a three channel RGB image, increases the already significant amount of computational resources required to train a typical deep CNN.

Previous approaches to creating CNNs for multi-spectral imagery have focused on fine-tuning CNNs that were pre-trained on RGB images, typically ImageNet. Three band combinations had to be selected manually from multi-spectral images due to the architectural constraints of the original RGB network. If a new three band combination was desired, the network had to be re-trained [5–7]. This approach discards a significant portion of the spectral information present in a multi-spectral image, almost defeating the purpose of acquiring high dimensional imagery in the first place. Other approaches that attempt to use more than three spectral bands often employ complex, multi-stage strategies to separate individual bands, reduce the dimensionality of the spectral bands, then perform some form of concatenation of the transformed spectral bands for later stages of a classifier [8, 9]. If we could have a small network architecture that could be trained from scratch on multi-spectral images, we could extract features from all spectral bands simultaneously, more effectively utilizing the information present in all of the available spectral bands, while at the same time keeping computational costs low. A small network with few parameters would also reduce the risk of overfitting to the smaller labeled datasets that are common with high dimensional imagery.

In this work we adapt recent developments in compressed, efficient convolutional networks (e.g. MobileNets [10] and SqueezeNets [11]) to create a small (in terms of number of parameters) and computationally efficient convolutional neural network, capable of being trained to process the full spectral-spatial input space from multi-spectral satellite imagery. We show that it is possible to train such a network end-to-end on multi-spectral imagery, using no data augmentation, that achieves greater classification accuracy and higher sample

efficiency than an identical network trained on RGB images.

We extend our analysis to a Bayesian version of the CNN architecture in question, using Monte Carlo dropout [12], and show that with a multi-spectral network, epistemic uncertainty in the model is significantly reduced compared to an identical network using 3-channel RGB images. This allows us to be more confident in the predictions a multi-spectral network is making. This is the first analysis that we are aware of comparing the uncertainty characteristics of RGB images vs. multi-spectral images.

The main contributions of this work are:

1) Demonstrating that small CNN architectures can be extended successfully to high dimensional imagery. This is in contrast to the large and complex models used in other studies.
2) Analyzing the performance of depthwise separable convolutions when applying CNNs to high dimensional imagery
3) Analyzing the uncertainty associated with using a deep neural network to make predictions with high dimensional multi-spectral imagery vs. using RGB images.

II. Related Work

CNNs have been used to classify individual pixels in remote sensing imagery by performing 1-D convolutions on the spectrum of each pixel [13]. In [9] this work was extended to be a spectral-spatial approach by applying 1-D convolutions on a flattened vector formed from the spectra of a group of adjacent pixels. None of these approaches maintains the physical distribution of spatial features when performing inference; the shape of spatial features is completely lost.

More recent work on the application of CNNs to satellite and multi-spectral imagery incorporated spatial context and generally focused on transfer learning approaches. Pretrained networks, typically using ImageNet as the training dataset, were fine tuned to work with satellite imagery by removing the final one or two layers from the pretrained network and then new final layers were inserted and trained from scratch on a small amount of labeled satellite imagery [14].

In [5–7], the aforementioned transfer learning technique was explored using RGB imagery. Penatti et al. [6] investigated how well features extracted from images of everyday objects extended to remote sensing and aerial scenes. They also fine tuned a network pretrained on ImageNet to predict poverty estimates based on night time light intensity. Castelluccio et al. [7] performed similar work, applying two networks pretrained on ImageNet to aerial RGB images. These transfer learning approaches worked surprisingly well, given that the datasets the networks were trained on included classes of objects from entirely different perspectives than those obtained from high altitude overhead images.

Relatively little work has been devoted to applying CNNs to high dimensional multi-spectral imagery. In [14] multi-spectral images from the Sentinel satellite constellation were analyzed, again used a transfer learning approach. The authors used a Google LeNet and a ResNet-50 network, pretrained on ImageNet, to perform land cover classification. These networks were fine tuned to produce land cover classifications by replacing their final fully connected layers. The authors manually chose 3 band combinations as inputs to the network, since the architecture of the pretrained model could not be modified to accept more than three spectral bands. Zhao and Du [15] used compressed spectral features from a local discriminant embedding method that are concatenated with spatial features from a CNN and fed into a multi-class classifier. In [8] a relatively complex network was created that splits a high-dimensional image into separate channels that were then fed to individual sub-networks to learn band specific features, these features were then concatenated and fed into a series of fully connected layers for final classification.

III. Adapting Efficient Architectures

The primary difficulties in working with multi-spectral imagery lie in its high dimensionality and the lack of large labeled datasets for training. Both of these factors make training recent deep network architectures from scratch on multi-spectral imagery difficult, often to the point of being impractical. A modern deep architecture such as a ResNet [16], or DenseNet [17], contains enough parameters to easily overfit to a small dataset. The high dimensionality of multi-spectral imagery can also significantly increase computational requirements.

To overcome these issues, we adopt strategies that maximize accuracy with a limited budget of parameters. Several relevant strategies to this goal are presented in recent work on small and efficient convolutional neural networks (e.g. SqueezeNets [11] and MobileNets [10]). While both SqueezeNets and MobileNets were originally conceived as a way to reduce the number of parameters in convolutional neural networks and make them more efficient, we hypothesize that, in addition to improving efficiency, some of the techniques used in those architectures will also be effective for processing high dimensional imagery.

A. Parameter Reduction Strategies

The first parameter reduction strategy is to forgo placing a fully connected network on top of the convolutional layers. Instead we use the same approach as the original SqueezeNet, and use a final convolutional layer with the number of output filters equal to the number of classes. A global average pooling is performed over these final filters to produce a logit for each class. This approach was inspired by Lin et al. [18]. The second parameter reducing strategy is to incorporate a large number of 1×1 convolutions vs. 3×3 convolutions, as 1×1 convolutions have $9\times$ fewer parameters. The final parameter reducing strategy is to limit the number of input channels to the 3×3 filters, since 3×3 filters make up the majority of parameters in the network. As the total number of parameters in a convolutional layer made up of 3×3 filters is (number of input channels) $\times$ (number of filters) $\times$ (3×3), reducing the number of input channels to these layers can significantly reduce the total number of parameters in a network.

In the original SqueezeNet architecture "squeeze" layers were introduced to reduce the number of input channels to "expand" layers that consisted of a mix of 3×3 filters and 1×1 filters. The squeeze layers consisted entirely of 1×1 convolutions and were originally used as a parameter reduction strategy. We adapt the squeeze layer in our architecture to reduce the number of parameters in the network and to compress the high dimensional input. Since the squeeze layer consists entirely of 1×1 convolutions, it essentially computes linear combinations between spectral bands.

This strategy worked well with RGB images [11], but it is not obvious that these "squeeze" layers would perform well with high-dimensional images where rapidly collapsing the initial input represents a major information bottleneck. The challenge then is to balance this bottleneck effect with the number of parameters in the network. In an effort to offset the reduced spatial context incurred by a large number of 1×1 convolutions, we also reduce the amount of pooling taking place early in the network, compared to the original SqueezeNet architecture, to maintain large activation maps.

B. Depthwise Separable Convolutions

In order to reduce the computational cost of working with high dimensional imagery, we also developed a second version of our network architecture that incorporates depthwise separable convolutions. Depthwise separable convolutions are composed of depthwise and pointwise convolutions. As shown in Fig. 1, the depthwise convolutions apply a single filter to each input channel, after which a 1×1 pointwise convolution is applied to create a linear combination of the outputs from the depthwise filters. Depthwise convolution can be written as,

$$\hat{\mathbf{G}}_{k,l,m} = \sum_{i,j} \hat{\mathbf{K}}_{i,j,m} \times \mathbf{F}_{k+i-1,l+j-1,m}$$

where M is the number of channels, $\hat{\mathbf{K}}$ is the depthwise convolutional kernel of size $D_K \times D_K \times M$ and $\mathbf{F}$ is a set of filters. The m_{th} channel in $\hat{\mathbf{K}}$ is applied to the m_{th} filter in $\mathbf{F}$.

The computational cost of depthwise separable convolutions is the sum of the depthwise and pointwise convolutions,

$$O(MD_K^2 D_F^2 + MND_F^2),$$

where D_F is the spatial shape of a square input, M is the number of input channels, and N is the number of output filters. Regular convolutional filters have a cost of,

$$O(MND_K^2 D_F^2).$$

Thus, depthwise separable convolutions reduce computation by a factor of,

$$\frac{MD_K^2 D_F^2 + MND_F^2}{MND_K^2 D_F^2} = \frac{1}{N} + \frac{1}{D_K^2}$$

There is actually a significant amount of similarity between elements of the squeeze layers presented earlier, and depthwise separable convolutions. The pointwise convolutions in the depthwise separable convolutions are equivalent to the squeeze layers. They act to create linear combinations of input channels. Given this similarity, depthwise separable convolutions can be substituted for the squeeze layers in our CNN architecture.

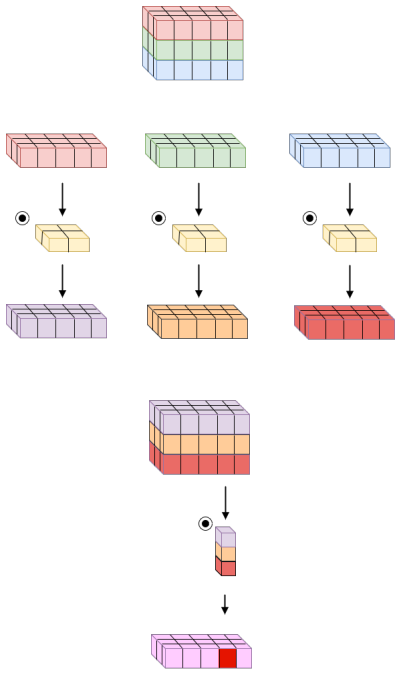

Fig. 1. Depthwise separable convolution.

In our first network architecture, a single convolutional kernel is used to extract cross channel correlations and spatial correlations simultaneously. The key difference between the first network architecture and this architecture which incorporates depthwise separable convolutions is the decoupling between spectral bands and the spatial correlations. The Xception architecture hypothesizes that cross-channel correlations and spatial correlations are sufficiently decoupled that it is actually preferable not to map them jointly [19].

It is not clear if this decoupling is preferable when dealing with multi-spectral imagery. In a RGB classification task like ImageNet, which has many classes that are easily separable even if the images are grayscale and contain no color information (e.g. space shuttle and soccer ball), simultaneously mapping the coupling between spectral and spatial correlations likely matters little. However, when working on a task where spectral information almost always provides useful additional information, such as land cover classification, it may be desirable to map spatial and spectral correlations together.

IV. The SpectrumNet Architecture

We are calling our network SpectrumNet, and it closely mimics the macro-architectural design of the SqueezeNet architecture [11] while adapting the micro-architectural elements to work well with high-dimensional imagery. The macro-architecture of the network defines "spectral" modules, which consist of the squeeze and expand layers described in the previous section. A depiction of the spectral module is

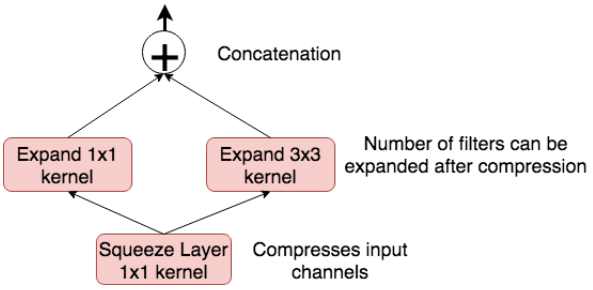

Fig. 2. A spectral module.

shown in Fig. 2. This type of macro-architectural element was referred to as a "fire" module in [11].

The primary adaptations we made were to further limit the number of 3×3 convolutions in the network, by setting the percentage of 1×1 convolutions in the network to 75%, in addition to delaying any pooling operations until midway through the network, with one final max pooling operation before the final spectral module. These settings were tuned through a series of experiments on our validation set. Our adapted SpectrumNet architecture is shown in Table I.

The second version of our network incorporates depthwise separable convolutions into the squeeze and expand layers of the spectral module. Table II details the reduction in network parameters and multiply-add operations that can be achieved with such an architecture.

V. Uncertainty Quantification

The probabilities given by a softmax layer at the output of a neural network are often wrongly construed as the confidence a model has in its output. It is entirely possible for a model to have a high softmax output and high uncertainty in that output. Principled uncertainty estimates can provide crucial information, allowing us to handle uncertain network outputs and be more confident when deploying deep learning models. As an example, on a classification task we could forward images that a model is highly uncertain about to a human for review.

Recent work by Gal and Ghahramani, has presented dropout as a Bayesian approximation of Gaussian processes. We refer the reader to [12] for a derivation showing that the dropout objective is mathematically equivalent to an approximation of a probabilistic deep Gaussian process. The basic idea is that both dropout and Gaussian processes place distributions over random models or functions.

Model uncertainty can be obtained from any network architecture that uses dropout. Following the results presented in [12], our approximate predictive distribution is given for a new input point $\mathbf{x}^*$ by,

$$q(\mathbf{y}^*|\mathbf{x}^*) = \int p(\mathbf{y}^*|\mathbf{x}^*, \omega) q(\omega) d\omega$$

where $\mathbf{y}^* \in \mathbb{R}^D$, D is the number of classes, and $\omega = \{\mathbf{W}_i\}_{i=1}^L$ is the set of random variables in a network with L layers.

We can estimate the mean and variance of the approximate predictive distribution empirically by performing T forward passes through our model with dropout and collecting the set of vectors created by the Bernoulli distribution introduced by dropout, $\{\mathbf{z}_1^t, ..., \mathbf{z}_L^t\}_{t=1}^T$, with $\mathbf{z}_i^t = [z_{i,j}^t]_{j=1}^{K_i}$, and K_i defining the dimension of the weight matrix for a particular layer, giving $\{\mathbf{W}_i^t, ..., \mathbf{W}_L^t\}_{t=1}^T$.

The above procedure has been shown to provide a Monte Carlo estimate of the predictive mean and predictive uncertainty, and it is referred to as Monte Carlo dropout in the original derivation. In practice, we perform T stochastic forward passes through the model and calculate the mean and variance of the output distribution to obtain an estimate of uncertainty.

With this technique in hand and a network architecture that is capable of processing the full spectral-spatial input from high dimensional imagery, we can evaluate how the uncertainty of deep neural networks depends on the information contained in spectral channels, which to our knowledge, has not been explored previously.

VI. Experiments

We investigate how accuracy, convergence rate, and prediction uncertainty change using RGB images vs. using multi-spectral images. Additionally we evaluate how accuracy changes with reductions in the size of the training set.

For all experiments, the networks with standard convolutions were trained using stochastic gradient descent with Nesterov momentum set to 0.9 and weight decay set to $5e-4$. When using depthwise separable convolutions, we did not use weight decay due to the small number of parameters in the network. The initial learning rate was set to 0.001 and was reduced by 25% every 10 epochs. A batch size of 64 was used, and all experiments were run on a Nvidia GTX 1080TI GPU. All results presented are the result of 10 fold cross-validation experiments.

A. Dataset

We conducted our experiments on the EuroSat land cover classification dataset, which was compiled from Sentinel satellite imagery [14]. This is one of the largest multi-spectral datasets that has been created, consisting of 27,000 images from 10 different land cover classes: annual crop, permanent crop (e.g. fruit orchards, vineyards), forest, herbaceous vegetation, highway, pasture, river, sea/lake, industrial, and residential. Fig. 3 shows an example of imagery from a Sentinel satellite.

The spatial resolution of the spectral bands is either 10 meters/pixel or 20 meters/pixel. Bands with a lower spatial resolution were upsampled to 10 meters/pixel using cubic spline interpolation. The Sentinel satellites acquire 13 spectral bands, but only 10 are used in these experiments as 3 of the bands are low resolution (60 meters/pixel) and used for detecting things like aerosols or water vapor in the atmosphere.

TABLE I
THE SPECTRUMNET ARCHITECTURE.

Layer name/type	**Output size/filters**	**Filter size/stride** (if not a spectral module)	1×1 **Squeeze**	1×1 **Expand**	3×3 **Expand**
Input	$64 \times 64 \times 10$				
conv1	$32 \times 32 \times 96$	$2 \times 2/1$			
spectral2	$32 \times 32 \times 128$		16	96	32
spectral3	$32 \times 32 \times 128$		16	96	32
spectral4	$32 \times 32 \times 256$		32	192	64
maxpool4	$16 \times 16 \times 256$	$2 \times 2/2$			
spectral5	$16 \times 16 \times 256$		32	192	64
spectral6	$16 \times 16 \times 384$		48	288	96
spectral7	$16 \times 16 \times 384$		48	288	96
spectral8	$16 \times 16 \times 512$		64	385	128
maxpool8	$8 \times 8 \times 512$	$2 \times 2/2$			
spectral9	$8 \times 8 \times 512$		64	385	128
conv10	$8 \times 8 \times 10$	$1 \times 1/1$			
avgpool10	$1 \times 1 \times 10$	$8 \times 8/1$			

TABLE II
NUMBER OF PARAMETERS AND MULTIPLY-ADD OPERATIONS IN SPECTRUMNET, RESNET, AND DENSENET.

Network	**No. Parameters** RGB/Multi-Spectral	**Million Mult-Adds** RGB/Multi-Spectral
SpectrumNet w/standard convolution	727.59k / 730.28k	203.47 / 206.22
SpectrumNet w/depthwise separable convolution	266.47k / 269.16k	75.59 / 78.34
ResNet-50	25.56M / NA	4120 / NA
DenseNet-161	28.68M / NA	7820 / NA

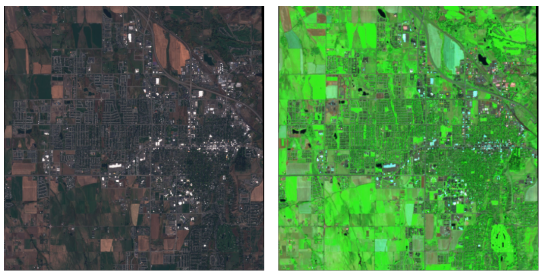

Fig. 3. A Sentinel image: RGB on the left and SWIR on the right.

B. Classification Performance

We evaluated both RGB images and 10 band multi-spectral images. The multi-spectral bands consisted of the RGB bands plus three red edge bands ranging from 705 nm to 783 nm, two near infrared bands at 842 nm and 865 nm, and two short wave infrared bands (SWIR) at 1610 nm and 2190 nm.

A network trained with multi-spectral imagery consistently achieved statistically significant higher classification accuracy compared to a network trained with RGB images (Table III). We expected multi-spectral images to result in greater classification accuracy in theory, given the additional information that is present in the additional spectral bands that increases separation between land cover classes. However, it was not clear if a small network architecture could effectively process such high-dimensional imagery. These experiments

TABLE III
CLASSIFICATION ACCURACY.

Network	**RGB Accuracy**	**Multi-Spectral Accuracy**
SpectrumNet w/standard convolution	$92.1 \pm 0.9\%$	$96.6 \pm 0.4\%$
SpectrumNet w/depthwise separable convolution	$87.6 \pm 0.8\%$	$95.2 \pm 0.7\%$

show that it is possible to use small and efficient CNN architectures to map and extract spectral-spatial features from high dimensional imagery simultaneously. Complex schemes built upon band separation, dimensionality reduction, etc. do not appear to be needed for this type of task.

Incorporating depthwise separable convolutions into our architecture resulted in slightly reduced accuracy compared to using standard convolutions. However given the significant reduction in computation requirements when using depthwise separable convolutions this may be a worthwhile trade-off. The solid performance of depthwise separable convolutions in this domain of high dimensional imagery lends further credence to the Xception hypothesis that spatial and spectral correlations can be mapped separately [19].

C. Sample Efficiency

We found that having a small network capable of effectively processing multi-spectral imagery resulted in faster convergence rates, and greater sample efficiency during training. As Fig. 4 shows, a network trained on multi-spectral imagery reaches convergence in $\approx$ 4,000 to 5,000 fewer gradient descent steps compared to using RGB images. This highlights an interesting, and significant result. Not only does using multi-spectral imagery result in greater classification accuracy, a CNN that can work well with multi-spectral imagery is also significantly more sample efficient.

As a further exploration of the sample efficiency of a small CNN applied to high dimensional imagery, we evaluated classification accuracy of our network trained on reduced

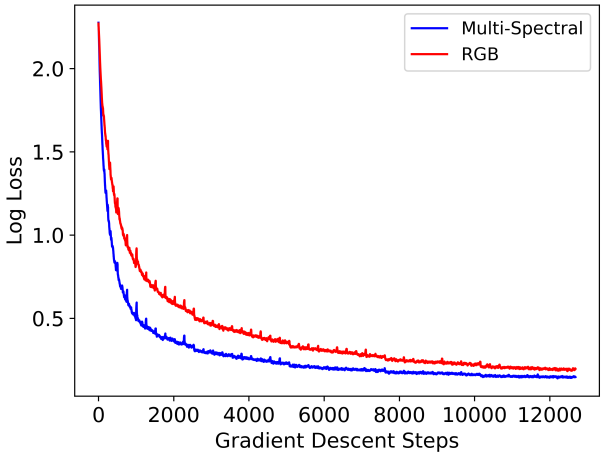

Fig. 4. Training profile for SpectrumNet with standard convolutions.

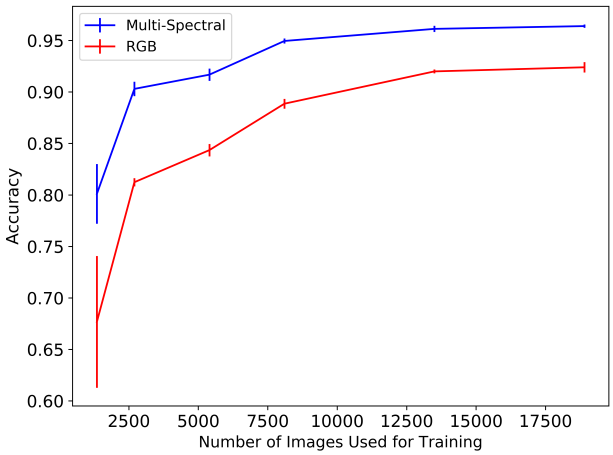

Fig. 5. Classification test set accuracy as the size of the training set is reduced.

datasets of both RGB and multi-spectral images. As the size of the training set was reduced, our small CNN architecture was able to maintain high accuracy ($\approx 90\%$) when the training set consisted of 2,500 images. When trained on RGB images, the accuracy of the same network architecture experienced a greater reduction in accuracy as the training set was reduced, compared to the network trained on multi-spectral images. At a training set size of 2,500 images, the multi-spectral network experienced a 6.1% reduction in classification accuracy, while the network trained on RGB images had experienced an 11.2% reduction in classification accuracy. This gap widened as the training set was reduced to 1,350 images, (Fig. 5).

While using depthwise separable convolutions resulted in similar classification accuracy as standard convolutions, convergence was appreciably slower (Fig. 6). This suggests that decoupling cross-channel correlations and spatial correlations makes it more difficult for the network to learn; although,

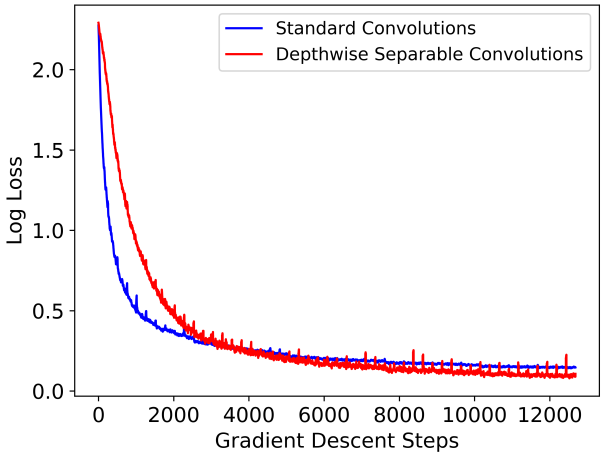

Fig. 6. Training profile using standard vs depthwise separable convolutions.

with sufficient training examples, comparable classification accuracy to standard convolutions can be achieved.

D. Uncertainty Quantification

We assessed network uncertainty characteristics for RGB images and multi-spectral images using Monte Carlo dropout, with 50 stochastic forward passes of our network. On average we observed much lower model uncertainty when using multi-spectral images. Fig. 7 shows an example of the distributions of the network outputs obtained for a single image using just the RGB bands of the image, compared to using 10 spectral bands from the image. In this case the image was correctly classified both when using RGB bands and 10 spectral bands. Although a correct classification was achieved using just RGB bands in this particular case, the multi-spectral image results in lower variance in network predictions, indicating that the additional spectral bands reduce the uncertainty in our network's predictions.

It is also interesting to observe the case where an image is incorrectly classified using RGB bands but correctly classified when using the full multi-spectral image. Fig. 8 illustrates this situation. In this case, class 0 was the correct class, and class 3 was the incorrect class predicted using just the RGB bands. When using the multi-spectral image, the separation between classes is large, and the variance associated with each class is small. In contrast, when using the RGB bands, the variance associated with each class is wide, to the point that the distributions of the network's outputs are overlapping. This reveals the high uncertainty associated with only using the RGB bands to make a classification decision in this case. On the rare occasions when an image class is predicted correctly using RGB and predicted incorrectly with multi-spectral, high variance is typically exhibited using both band combinations.

We also compiled class wide statistics on the distributions of our network's outputs for each land cover class, using RGB images and multi-spectral images. Fig. 9 shows a kernel density estimate of these distributions. The classes, "residential"

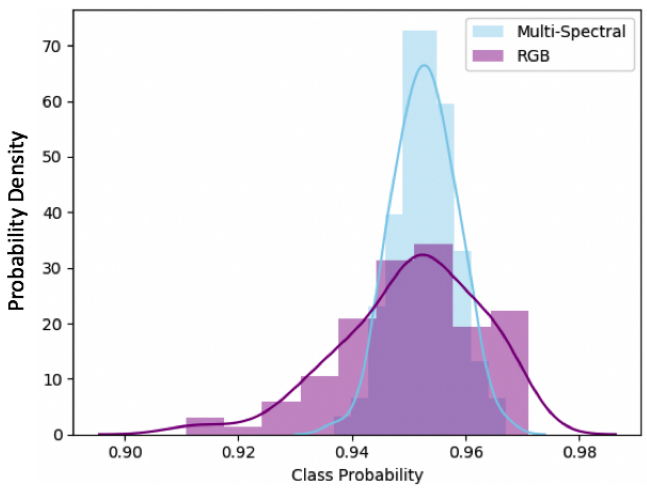

Fig. 7. Network uncertainty for an image that was correctly classified both when using RGB bands and multi-spectral bands.

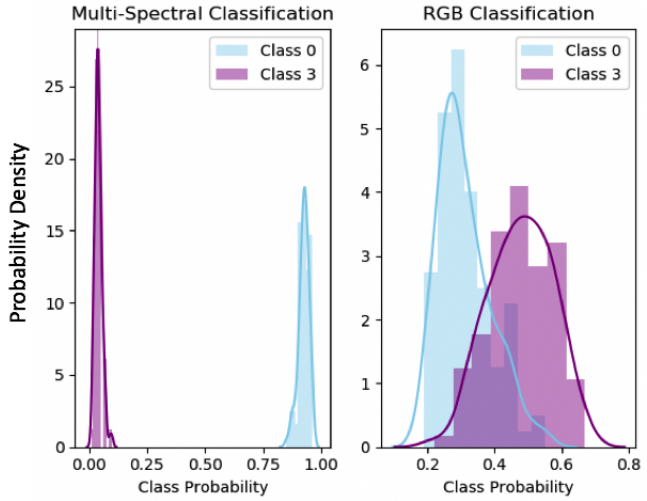

Fig. 8. Network uncertainty for an image that was incorrectly classified using RGB bands, but was correctly classified when using 10 multi-spectral bands. In this case class 0 is the correct class, and class 3 is the incorrect class predicted when using the RGB image.

and "industrial" are omitted, as both classes exhibited low uncertainty when using both RGB and multi-spectral images. Fig. 9 shows the substantial reduction in uncertainty that can be obtained when we have a network that works well with high dimensional imagery.

VII. Conclusion

Most previous approaches to dealing with high-dimensional imagery, and the typically small datasets associated with this type of imagery, have focused on transfer learning. These approaches have not utilized the additional spectral information offered by multi-spectral and hyper-spectral imagery effectively. Other techniques have employed complex multi-stage processes to separate and reduce the dimensionality of the input. In contrast to previous work, we showed that our CNN architectures, which have fewer than one million parameters, can be applied effectively to high dimensional imagery. We also demonstrated that our architecture works well with a small training set, making it particularly useful in working with the small datasets that are common when dealing with multi-spectral imagery.

Depthwise separable convolutions reduced the computational requirements of our network significantly, and we observed only small deterioration in classification accuracy due to the decoupling between spectral and spatial correlations. However, depthwise separable convolutions proved to be less sample efficient than standard convolutions in our experiments, suggesting that standard convolutions may be the preferred choice when working with small datasets.

Beyond higher classification accuracy, a major benefit of a CNN architecture that can process multi-spectral imagery is the significantly reduced uncertainty exhibited by the model, compared to a network that only uses RGB images. Therefore, we can be more confident when deploying models that use multi-spectral imagery. This is a previously unreported result in the literature surrounding remote sensing and deep learning and should further motivate the development of CNN architectures, not just for RGB images, but for high dimensional imagery as well.

Future work will be directed at confirming the results presented here with additional datasets and classification tasks. We are currently evaluating the effectiveness of our CNN architectures in classifying the health of produce in grocery stores using hyper- and multi-spectral imagery.

Acknowledgments

The authors thank our fellow members of the Numerical Intelligent Systems Lab at MSU for productive discussions, as well as, Riley Logan, and Bryan Scherrer for helpful information regarding multi-spectral and hyper-spectral imagery.

References

[1] European Space Agency. (2018) Sentinel online. [Online]. Available: https://sentinel.esa.int/web/sentinel/home

[2] H. M. Miller, N. R. Sexton, L. Koontz, J. Loomis, S. R. Koontz, and C. Hermans, *The users, uses, and value of Landsat and other moderate-resolution satellite imagery in the United States-Executive report*. US Department of the Interior, Geological Survey, 2011.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[4] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Citeseer, Tech. Rep., 2009.

[5] M. Xie, N. Jean, M. Burke, D. Lobell, and S. Ermon, "Transfer learning from deep features for remote sensing and poverty mapping," *arXiv preprint arXiv:1510.00098*, 2015.

[6] O. A. Penatti, K. Nogueira, and J. A. dos Santos, "Do deep features generalize from everyday objects to remote

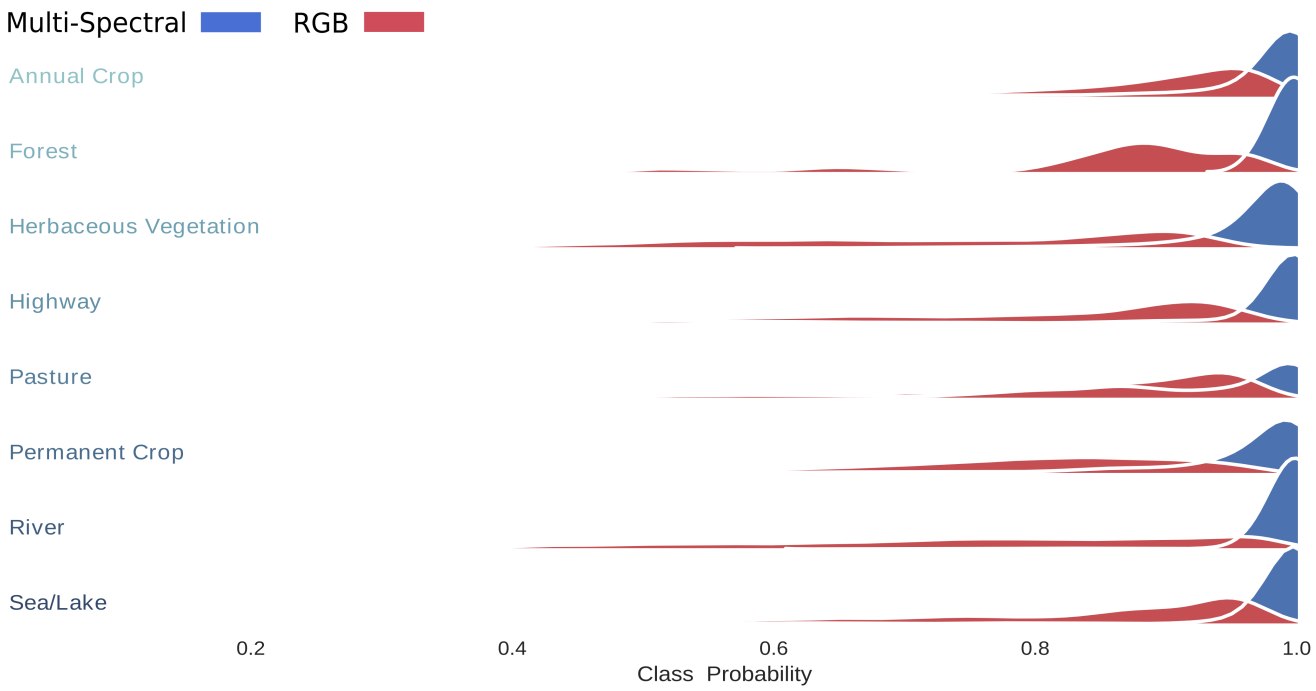

Fig. 9. Network uncertainty using RGB vs. multi-spectral images partitioned by land cover class. The distributions displayed are kernel density estimates computed from histograms.

sensing and aerial scenes domains?" in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2015, pp. 44–51.

[7] M. Castelluccio, G. Poggi, C. Sansone, and L. Verdoliva, "Land use classification in remote sensing images by convolutional neural networks," *arXiv preprint arXiv:1508.00092*, 2015.

[8] A. Santara, K. Mani, P. Hatwar, A. Singh, A. Garg, K. Padia, and P. Mitra, "Bass net: band-adaptive spectral-spatial feature learning neural network for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 9, pp. 5293–5301, 2017.

[9] Y. Chen, X. Zhao, and X. Jia, "Spectral–spatial classification of hyperspectral data based on deep belief network," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 6, pp. 2381–2392, 2015.

[10] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[11] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and $<$ 0.5 mb model size," *arXiv preprint arXiv:1602.07360*, 2016.

[12] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*, 2016, pp. 1050–1059.

[13] V. Slavkovikj, S. Verstockt, W. De Neve, S. Van Hoecke, and R. Van de Walle, "Hyperspectral image classification with convolutional neural networks," in *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 1159–1162.

[14] P. Helber, B. Bischke, A. Dengel, and D. Borth, "Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification," *arXiv preprint arXiv:1709.00029*, 2017.

[15] W. Zhao and S. Du, "Spectral–spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 8, pp. 4544–4554, 2016.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[17] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks." in *CVPR*, vol. 1, no. 2, 2017, p. 3.

[18] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.

[19] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *arXiv preprint*, pp. 1610–02 357, 2017.