Land Use prediction in the Chesapeake Bay Watershead

Daniel Morton — SUID 06511962
dcmorton@stanford.edu

Abstract

Manual land cover mapping is a tedious and expensive process. Automation has met with limited success because land use does not look the same from one location to another. This paper will explore options for augmenting data to improve land use prediction over a wide area, including using multiple images from the same locations, infrared data, and prior knowledge of the region.

1. Introduction

Manual land cover mapping is a tedious and expensive process. To date, attempts to automate have met with limited success; usuallly failing to generalize well from one region to another. In simple terms, when the distribution of the labels changes, model accuracy suffers. Recently, there has been success using older and cheaper low resolution images and labels to improve prediction accuracy for the high resolution images. The techniques are often combined with, at this point, relatively primitive image segmentation architecture. In this paper we will combine state of the art deep learning networks for image segmentation with several preprocessing strategies to get near state of the art results. This work will be done using satelite data from the Chesapeake Bay region, which extends from New York and Pennsylvania down to Virginia and includes Maryland, West Virginia, and Delaware but not New Jersey.

In the interest of making the model as general as possible, we will use the smallest training set practical, namely the state of Delaware. The base model will be trained and validated on Delaware data. We will then extend the validation set to cover all states, and the training data to include multiple years, infrared data, and a lower resolution set of categories from an earlier dataset.

2. Related Work

Much of the work is this paper was inspired by **??**, which uses data from multiple sources and basic U-Nets to predict land use in the Chesapeake Bay region. The same authors have constructed a semi-supervised system for learning land iteratively. **??** There has also been work on using one set of labels to inprove the prediction of another set of labels (i.e. using priors) in image segmentation. **??** In that paper, normally distributed priors are used, wheras here we use a multinomial distribution.

3. The Data

The primary data consists of two sets of high resolution satelite images from the US Department of Agriculture's National Aerial Imagery Program]refUSGA covering the five state Chesapeake Bay watershead region. The two sets of data were collected in 2011/2012 and 2013/2014 and consist of both visual and near infrared images of the region. High resolution category labels over six categories **??** (water, forest, field, barren, impervious, and road) are also provided; these are the labels to be predicted from the image data. Supllimenting this are low resolution, but finer grained, land category **??**, which can be used as prior estimates, and low resolution Landsat data **??**, which was ultimately unused in this project.

The high resolution data is at 1m spacial resolution; the low resolution data is at 30m resolution, but has been reprojected to 1m (via upsampling) for consistency.

The full dataset is quite large; consisting of 125 tiles from each state (except Delaware which only has 107) each covering $45km^2$. For each of use small $256m \times 256m$ patches have been created from all the data sources. Five hundred patches have been sampled from each tile, which makes the patches a very representative subset of the orginal tiles. Tiles, and thus patches, have already been assigned to training and validation sets. It will become clear later that training and validation distributions are as similar as can be

Table 1. Category Percentage by State

	Water	Forest	Field	Barren	Impervious	Road
Delaware	2.1	38.7	51.6	0.5	4.8	2.4
Maryland	11	43.3	39.7	0.3	4	1.6
New York	3.7	59.5	34.6	0.1	1	1.2
Pennsylvania	0.9	66.8	27.3	0.5	3.2	1.2
Virginia	4.5	68.9	23.3	0.1	1.8	1.4

expected. There are 2500 validation patches for each state and 5000 training patches for each state except Delaware, which has 41000.

The distribution of surface categories, including the presence of water and land use, is heterogenous among the states. Road cover varies from 1.1% in New York to 2.4% in Delaware while forest covers more than two thirds of Virginia and Pennsylvania but less than half of Delaware and Maryland. Maryland is over 10% water, while Pennsylvania is less than 1%.

Land use changes over time, albeit slowly. The land use labels apply to the 2013/2014 data, but it is assumed that the same categories hold for the 2011/2012 data. This allows for a certain amount of data augmentation via use of two sets of images for each location.

There are 15 low resolution land use labels: open water, four levels of developed land based, baren land, three types of forest, shrubland, grassland, pasture, cropland, and two types of wetland. Although much finer grained than the high resolution categories each pointof low resolution labeling corresponds to 900 points of high resolution labeling, and thus refelect an average of all land use in that area.

Figure 1. 2011 Scene

Figure 2. 2013 Scene

4. The Model

The goal is to predict pixel level land use from image data using as small a training set as practical. We want a model trained on data from one state that can predict land use in another state. Ideally, this should only require visual data, but supplimentary data may be required.

The model has to be a neural network. The standard meta-architecture for this sort of problem is a U-Net, **??** an architecture that contains an encoder and a decoder component. The encoder consists of a series of blocks of convolutional layers of decreasing width and increasing depth while, in the original verison, the decoder was the same types of layers in reverse with skip connections between blocks of layers of the same size. The deeper and narrower layers can be seen as learing more complicated larger resolution features, while the earlier, shallower layers learn smaller finer grained features. In the decoder step, these low resolution and high resolution features are combined to predict a final ouput.

It was subsequently realized that the abosolute symmetry between the encoder and decoder were unnecessary. It is now possible to use standard image classification backend architectures as the encoders. The most powerful encoder architetures availablle at this time are the EfficientNet **??** family. Most experiments will be performed with the smallest member of this family, EfficientNetB0, with successful models extended to EfficientNetB1 and EfficientNetB2. The decoders will remain the same throughout. The final layer will produce softmax predictions for each pixel. The learining rate will start at 0.00256 and decay by a constant rate each step so that it decays by a factor of 0.94 every epoch. The adam optimizer is used. Models are trained for 50 epochs, but only models where valiation accuracy improves are saved. The model with the best validation score can then be recovered easily for running validation metrics.

Since the models are purely convolutional, they can accept inputs of any size. Full 256×256 patches are used for training, with no data augmentation beyond

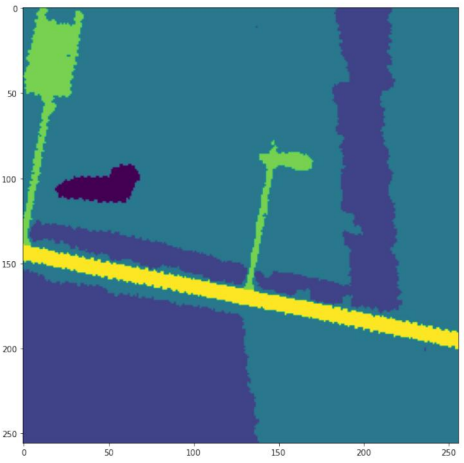

Figure 3. Land Use Categories

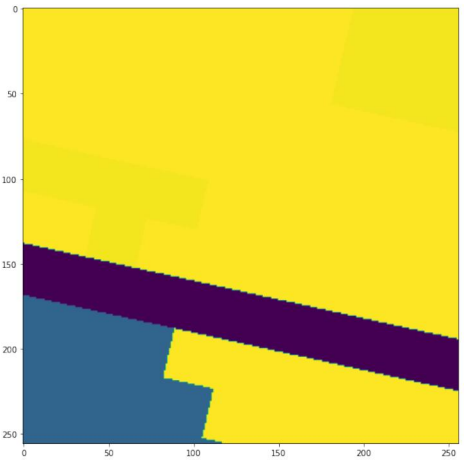

Figure 4. Low Resolution labels

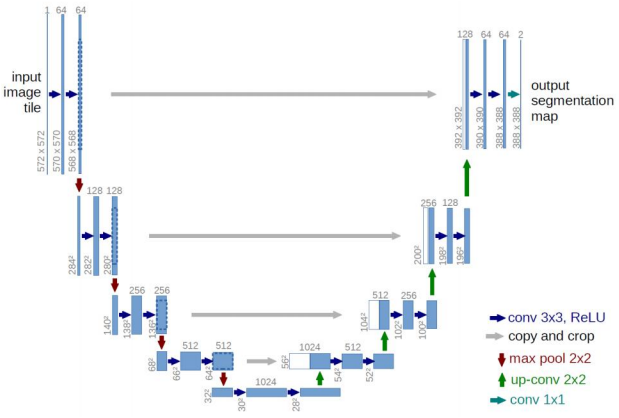

Figure 5. Original U-Net

centering and scaling the input.

Transfer learning is a deep learning strategy where weights trained on one task are reused as the initial weights for a second task. This works because the features learned by the first task, especially those of early

Table 2. Basic Models by State

	DE	MD	NY	PA	VA	WV	Tot
Pretrained	94.5	84.7	90.6	88.6	74.2	58.2	81.8
Not Pretrained	94.1	81.8	77.4	81.6	72.4	62.2	78.2

layers, are generally applicable to other tasks. The standard set of pretrained weights comes from training on ImageNet; it is not clear a-priori that these features would be helpful in learning images. It will turn out to be an important consideraton.

5. Experiments

The state with the smallest amount of data is Delaware. In the interest of producing the most general model, we train exclusively on the Delaware data set. Validation is then done on the validation sets for each of the states. The basic model would train on the Delaware train data and validate on the Delaware validation data. This could be done with or without ImageNet weights. The initial results were quite good, with 94.5% validation accuracy with ImageNet weights and 94.1% accuracy with no pretrained weights. But the model did not generalize well; with pretrained weights the overall accuracy was only 81.8% pretrained and a mere 78.2% with pretrained.

The first thing to notice is that the pretrained weights improve accuracy significantly (except in West Virginia) with the biggest jump in accuracy coming in New York. Nonetheless, accuracy has dropped by several percentage point everywhere and by double digits in most states.

It turns out a simple change can improve the model's ability to generalize. Models overfit when they perform better on the training set than on the validation set. In this case, the training accuracy on Delaware was 96%, only slightly better than the validation accuracy. But the training and validation sets had already been chosen well enough that similar performance on both sets was expected. In short, the model was overfitting on the Delaware validation set. The solution was obvious, use the validation data of all six states to get the model to stop training before overfitting becomes significant. With the new validation set putting the breaks on overfitting, accuracy increases to 86.8% for the pretrained network and 84.0% for the network with randomly initialized weights. There is a modest drop in accuracy for Delaware itself, but that is amply compensated for in the improved performance of Virginia and West Virginia. Maryland, Pennsylvania and New York only improved modestly, suggesting that they were already close to as accurate as the model could make them.

The second improvement in accuracy comes from

Table 3. Revised Validation Models by State

	DE	MD	NY	PA	VA	WV	Tot
Pretrained	94.0	84.8	91.3	89.1	79.9	82.2	86.9
Not Pretrained	93.1	85.8	84.7	89.7	76.3	74.4	84.0

Table 4. Two Year Models by State

	DE	MD	NY	PA	VA	WV	Tot
Pretrained	93.7	87.8	91.9	90.6	83.2	78.0	87.5
Not Pretrained	94.0	86.0	89.4	89.8	81.9	82.5	87.2

Table 5. Prior Models by State

	DE	MD	NY	PA	VA	WV	Tot
Prior Alone	80.4	79.5	84.7	82.1	84.6		
Prior and Model	93.6	87.6	90.6	90.2	81.2	80.6	87.3

Table 6. Two Year Models With and Without NIR

	DE	MD	NY	PA	VA	WV
Color Only	94.0	86.0	89.4	89.8	81.9	82.5
With TIR	94.1	88.8	90.9	90.2	66.6	63.5

expanding the training data. Land use changes slowly over time, but the appearance of land can change rapidly, sometimes in a matter of days. Even after an event a simple as a rainstorm any vegitation is likely to be greener than it had before. In order to given the model a better sense of what each class of land looks like, we use both satelite passes as training data. There are two possible ways to do this; either put both images through the network at the same time (using an imput tensor with six channels) or randomly select an image from each dataset while training. The first strategy is undesirable for two reasons; it can't be used with pretrained weight and it creates a model that would always need two images for input. The second strategy is the one we use. The end result almost closes the gap between the pretrained and non-pretrained models, with 87.5% accuracy on the pretrained model and 87.2% accuray on the randomly initialized model. It seems that the second set of data does not just produce a more robust model, but mimics the benefits of pre-trained weights as well.

5.1. Priors

The coarse grained categories can serve as priors to help improve accuracy. IIf I is the image, c is the fine grained category being predicted and C is the coarse grained category then it is a simple matter to compute $p(c|l)$. This is done by taking counts of each pair of categories (c, l)) over all pixels on the training tiles (not the patches) and dividing by the totals for each category l. We can even use the probabilities $p(c|l)$ to predict the fine-grained categories by the simple expedient of taking $\arg\max_c p(c|l)$. Even though the coarse grained categories are at a much lower resolution, the accuracy of this simple predictor is 80−85%, depending on the state. (West Virginia is missing because of a numerical overflow error) In this case, though, accuracy is deceptive; the naive Bayes classifier can only predicts the first three categories (water, field, and forest), but these categories dominate.

For better accuracy, the priors can be used a regularization term in the model loss function.

$$p(c|I, C) = \frac{p(c, I, C)}{P(I, C)} = \frac{p(C|I, c)p(c|I)}{p(C|I)}.$$

We can thus replace our original loss function (categorical cross entropy) with categorical_crossentropy + $\lambda KL(C||c)$ where KL is the Kulback-Leibler divergence. We tried three values of λ, 1, 10^{-1}, and 10^{-2}. Only the 2013/2014 data was used for training in this case, and only the model with pretrained weights The middle value performed best, with an accuracy of 87.3%, almost as good as the model using both sets of images.

5.2. Adding Infrared

When adding a feature in linear regression, even it proves to be simply noise, the new model always performs better on the training set. For the same reason, if we add a new channel of features to a neural network model we should expect that model to perform better on the training set. And just as in the case of linear models, we should not expect that improved performance to automatically carry over to validation. This exactly what happens when we add the NIR channel to the our model. There is a tiny increase in accuracy for Delaware, modest but respectable increases for Maryland, New York, and Pennsylvania, but performance collapses for Viriginia and West Virginia. This is likely due to climate; heat patterns in Delaware are closer to those in the northern Chesapeake bay states than they are to those in the southern Chesapeake region. The end result is that NIR confuses the model in those states.

5.3. Expanding the Model Size

The results of this section should surprise no-one. With successful models trained on EfficientNetB0, it was natural to see if larger versions of the same architectures work better. The EfficientNet family is constructed by increasing the number of layers and the width (number of filters per layers) as well as the resolution of the input images in a systematic way. In this case image resolution is fixed, but we would still expect a strictly larger model to perform better. The

Table 7. Larger EfficientNet Models

	DE	MD	NY	PA	VA	WV	
B0 - Two year	93.7	87.8	91.9	90.6	83.2	78.0	87.5
B1 - Two Year	94.1	87.7	89.0	90.9	84.0	84.7	88.4
B2 - Two Year	94.1	88.1	89.3	91.3	87.0	87.8	89.6

Table 8. Prior Models by State and Size

	DE	MD	NY	PA	VA	WV	Tot
Prior Alone	80.4	79.5	84.7	82.1	84.6		
B0 with Prior	93.7	87.8	91.9	90.6	83.2	78.0	87.5
B2 with Prior	94.2	86.3	90.3	89.0	83.3	75.6	86.5

Table 9. Combined Models by State and Size

	DE	MD	NY	PA	VA	WV	Tot
B0 with Prior	93.6	87.6	90.6	90.2	81.2	80.6	87.5
B0 Two year	93.7	87.8	91.9	90.6	83.2	78.0	87.5
B0 Prior and Two Year	93.4	87.1	90.4	90.4	86.0	85.7	88.8
B1 Prior and Two Year	94.1	88.0	89.4	90.5	86.2	89.7	89.7

best models have been the models using two years of data and those using prior weights.

In the case of the two-year models, we see modest to significant increases in most states with very modest backsliding in Pennsylvania.

The same does not hold true for the models with the coarse grain label priors. In this case there is a modest decrease in accuracy. The decrease is small which means It is possible that the regularization weight of 10^{-1} is not optimal. EfficientNet B1 was skipped for time.

5.4. Combining Strategies

All the strategies so far proposed have been independent of each other. It is natural to build a model that combines them. The end result is interesting. Although there is usually a slight decrease in accuray for the high performing states (at this point anything that doesn't have Virginia it it's name), the two low performing states (Virginia and West Viriginia) now have accuray in line with their northern counterparts. It seems that the priors help most in cases where the images have less predictive power, but have a slight flattening effect on predictions in cases where the the images alone perform well. EfficientNet B2 was skipped for time.

5.5. Accuracy by Category

Up to this point, nothing has been said about the accuracy by category of the models. For this, we only consider three states and three models. Surprisingly, the model with priors does not perform as well on all classes as base model (although barren land, the least common class is the only one to be seriously hurt), and simply using two sets of input data does more to improve accuracy for the less common classes. We also see a key reason why Virginia has had low accuracy;

Table 10. Class Level Accuracy

	Water	Forest	Field	Barren	Impervious	Road
DE Base Model	88.7	94.4	96.6	49.1	77.8	82.1
DE Two Year	88.0	93.7	96.2	43.5	74.4	76.0
DE Two Year Prior	86.7	95.5	96.1	16.9	71.6	73.8
NY Base	91.2	89.8	92.2	53.2	72.0	57.3
NY Two Year	96.2	90.5	93.8	53.6	72.4	56.7
NY Two Year Prior	86.6	89.1	94.8	8.0	70.6	49.9
VA Base Model	9.4	73.5	90.4	19.2	56.3	32.1
VA Two Year	44.1	84.1	89.9	0.9	66.1	46.0
VA Two Year Prior	8.6	90.2	90.6	0.2	59.0	38.4

the model has been unable to recognize water features.

6. Conclusions and Future Work

At this point the best models perform just under 90% across all the state. We have shown that expanding the validation set to include a representative sample of all the regions we want to predict for is a good way to combat overfitting and improves overall accuracy even when the training data is not changed at all. This does require going beyond the distribution of the original training data, but it can be accomplished with a small subset of the data we ultimately want to predict on.

The models themselves make a difference. Earlier work was done with primitive U-Nets and had less than 80% accuracy without a number of augmentation strategies. Surprisingly, using pretrained weights helped immensely; even though the data is not at all similar to that found in ImageNet, the features learned from transfer learning prove valuable in analyzing satelite data.

Adding a second set of satelite images gave the model a more robust understanding of what each category looked like, which improved accuracy. More valuable was the addition of the lower resolution categories; especially in states (Virginia and West Virginia) where a purely visual model underperformed.

Surprisingly, adding infrared data did not help the model at all, underscoring the limits of using image data to make predictions.

The obvious first step in any future work would be to extend the model to larger EfficientNets. This should be done in conjunction with replacing the patches currently being used with larger samples taken from the original tiles. This would allow for the application of some standard image detection preprocessing. Little work was done to standardize the color distributions across images; which would likely have improved accuracy as well. The low resolution Landsat data remains as a possible extra data source, although it is looking redundant at this point. It seems likely that the loss function between the prior estimates and cross-entropy is not yet optimal.

References

[1] Chesapeake Conservancy. Land cover data project.

[2] Homer C, Dewitz J, Yang L, Jin S, Danielson P, Xian G, Coulston J, Herold N, Wickham J, Megown K. 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. May 2015

[3] Malkin K, Robinson C, Hou L, Soobitsky R, Czawlytko J, Samaras D, Saltz J, Joppa L, Jojic N. Label super-resolution networks. International Conference on Learning Representations (ICLR). 2019.

[4] Robinson C, Hou L, Malkin K, Soobitsky R, Czawlytko J, Dilkina B, Jojic N. Large Scale High-Resolution Land Cover Mapping with Multi-Resolution Data. Computer Vision and Pattern Recognition (CVPR). 2019.

[5] Robinson C, Ortiz A, Malkin K, Elias B, Peng A, Morris D, Dilkina B, Jojic N. Human-Machine Collaboration for Fast Land Cover Mapping. arXiv 1096.04176, June 2019.

[6] Ronneberger, Olaf; Fischer, Philipp; Brox, Thomas (2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation".

[7] Tan, M. and Le, Q. V. Efficientnet: Rethinking model scaling for convolutional neural networks.

[8] United States Department of Agriculture. National Aerial Imagery Program.

[9] United States Geological Survey. Landsat 8.

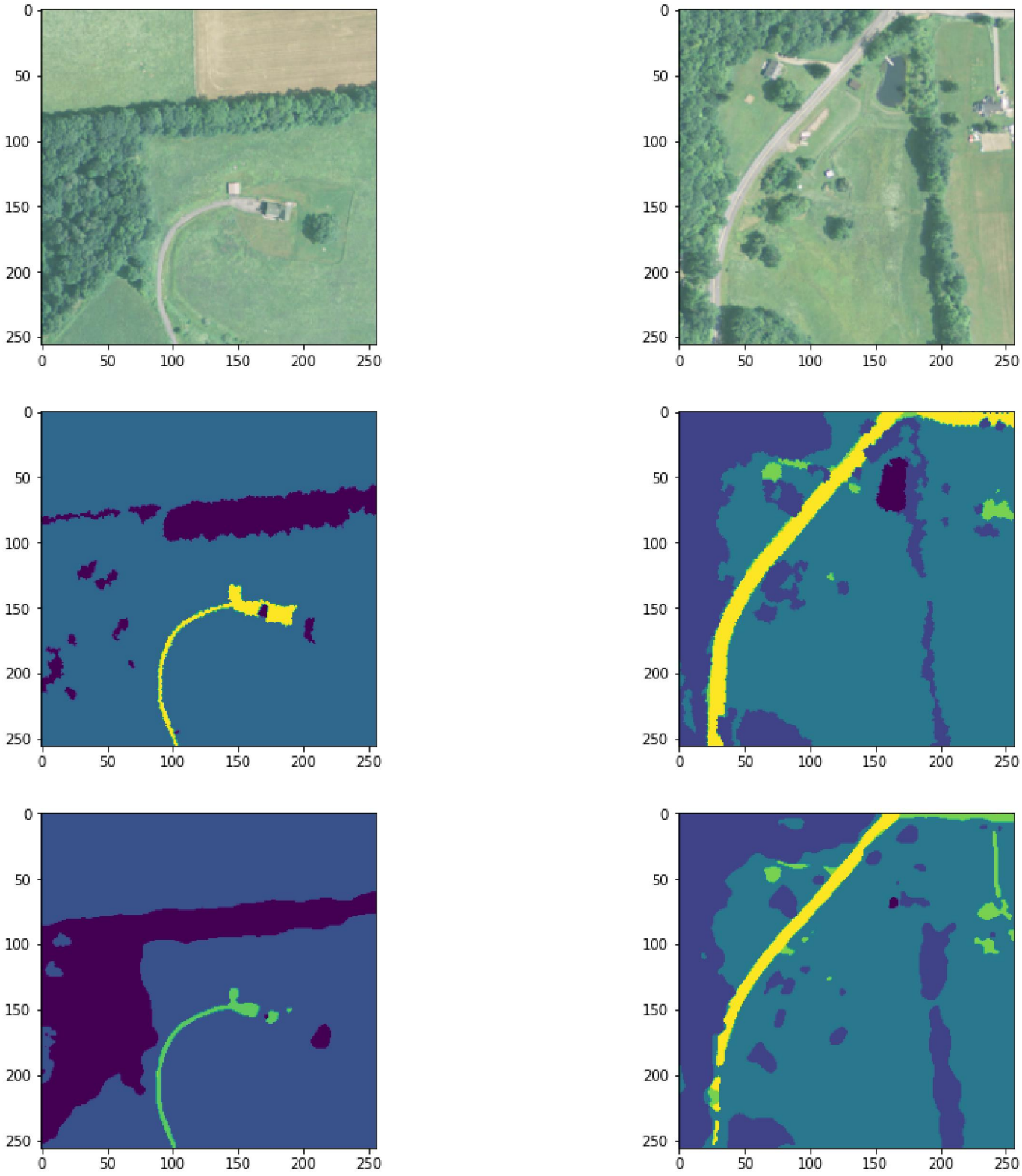

Figure 6. Two New York Scenes with Two Year (middle) and Two Year with Prior (bottom) Predictions. The middle row does a better job predicting water, while the bottom row is better at predicting forest.