

Final report: Land Cover Change Detection using Neural Network for Satellite Images

Ashkan Bozorgzad (ab5243), Hari Prasad Renganathan (hr2514), Karveandhan Palanisamy (kp2941),
Masataka Koga (mk4528), Yewen Zhou (yz4175), Yuki Ikeda (yi2220)
12/18/2022

Problem Definition and Overview

This project is sponsored by JPMorgan Chase, an investment banking company. The goal of the project is to create high-resolution (1m / pixel) land cover change maps of a study area, the state of Maryland, USA, given multi-resolution imagery and label data. This project aims to provide an example of situations commonly found worldwide. Landcover labeling can be used to detect land cover changes. Land cover change detection is important in many fields, such as environment, economics, and geology. For instance, JPMorgan can use it to evaluate environmental activities, including tree planting, by companies for their Environmental, Social, and Governance (ESG) Investing.

In the field of earth observation, new images produce faster than high-quality, high-resolution labels. However, only old and low-resolution labels are available, for example, 30m National Land Cover Database (NLCD) in the United States or 500 m MODIS land cover available worldwide. Therefore, it is significant to investigate how machine learning can be used to build a model that predicts high-resolution change without having a lot of higher-resolution change data. According to past studies [1], [2], and [3], weakly supervised segmentation and automatic super-resolution labeling are possible. These studies constructed high-resolution label predictors for high-resolution input imagery using regional supervision. They labeled a large block of land with four target classes (Water, Tree Canopy, Low Vegetation, and Impervious).

Past studies mostly made a model with a few phases, where a model with an architecture appropriate for semantic segmentation at the latter phase utilizes the labels predicted by another one at the former phase as input labels, for weakly supervised segmentation. Hereafter, we call this model a chained model or X_Y model, where X is a model at the former phase and Y is at the latter one. Therefore, we tried to implement chained models, and, based on their evaluation, decided on a U-Net18_U-Net50 model, where U-NetZZ denotes a U-Net model with the encoder of ResNet-ZZ as its backbone. The architecture of the U-Net18_U-Net50 model is as Figure 1.

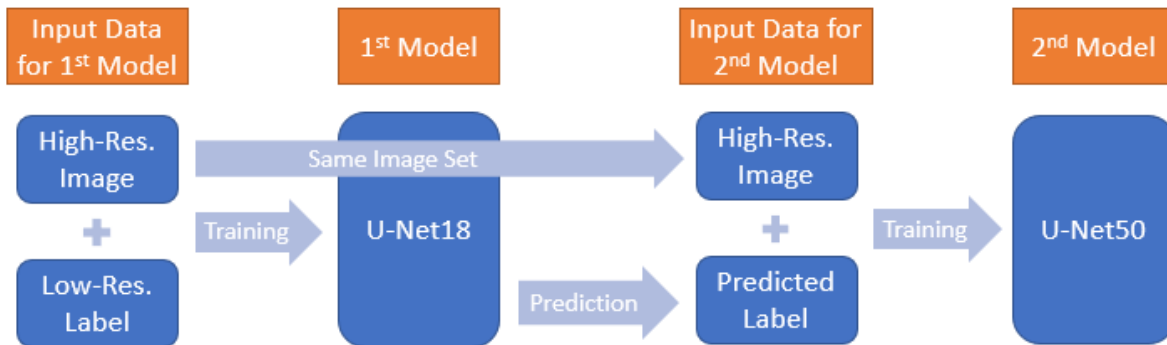


Figure 1 The architecture of our U-Net18_U-Net50 model

After creating a high-resolution model to predict land cover, predicting land cover change is a straightforward step. The model can be applied to two different years (images), and the super-resolved label can be compared to estimate changes in the four classes. The evaluation metric is the average intersection over union (IoU) between the eight classes' predicted and ground truth labels (loss and gain of the four target classes, except for no change). We can calculate the IoU by) below.

$$\text{IoU} = \frac{\# \text{ pixels labeled } c \text{ in the model's prediction and in the ground truth}}{\# \text{ pixels labeled } c \text{ in the model's prediction or in the ground truth}} \quad (\text{Equation 1}).$$

Also, we implemented a method to show reliable uncertainty level of our model prediction to each pixel, the Ensemble method [4]. We selected this method based on past studies, such as [5] and [6] which compared seven representative methods and concluded that the Ensemble method is one of the best methods appropriate for common machine learning tasks including image classification and semantic segmentation tasks. We made an ensemble model consisting of 10 U-Net18_U-Net50 models by selecting 10 input datasets randomly and training models with 10 different initial weights based on [6], which employed the Ensemble method for deep medical image semantic segmentation and showed that it outperformed another famous method, MC dropout. The overview of our overall approach is shown in Figure 2.

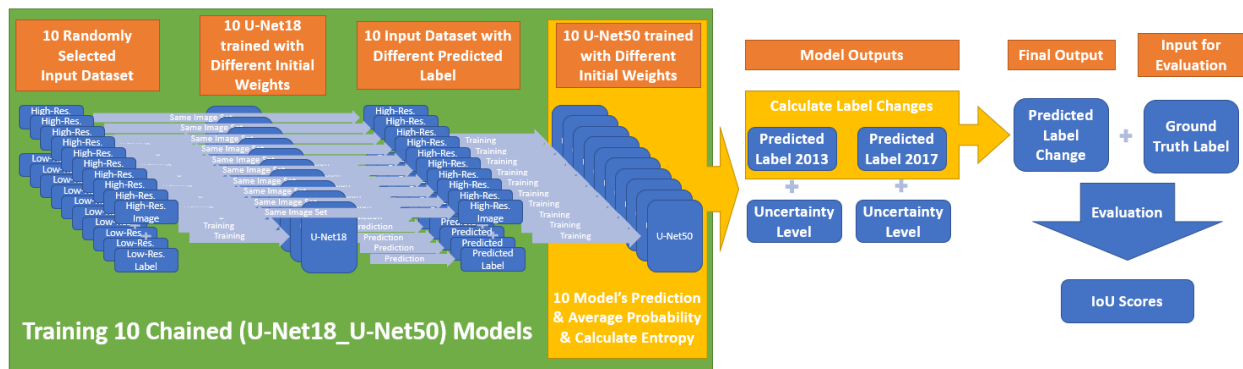


Figure 2 Summary of our overall approach

We also explored the use of newly published global land-cover label called Dynamic World label (DW), and investigated its performance as a model input for training instead of NLCD label. Since, DW is available globally (not just in US unlike NLCD) for anytime from 2015, it is of interest to investigate its performance, which may facilitate the use of DW for many regions and for latest years. Specifically, we first conducted exploratory data analysis for DW. Then, we trained 5-layer FCN model using either NLCD or DW and developed comparative analysis.



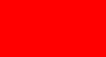









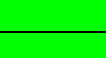
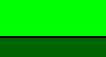

Dataset and Data Visualization

All data can be downloaded from the addresses listed at [Data Link](#). The input images comprise nine layers covering the state of Maryland in the United States (~35,000 Km²). All layers are upsampled from their native resolutions to 1m / pixel and provided as 2250 aligned tiles of dimensions not exceeding 4000 × 4000. The layers are as follows [7]:

1. NAIP (2 layers): 1m-resolution 4-band (red, green, blue, and near-infrared) aerial imagery from the US Department of Agriculture's National Agriculture Imagery Program (NAIP) from two points in time: 2013 and 2017.
2. Landsat (5 layers): 30m-resolution 9-band image from the Landsat-8 satellite from five-time points: 2013, 2014, 2015, 2016, and 2017. Each image is a median composite from all cloud and cloud-shadow-masked surface-reflectance scenes intersecting Maryland.
3. NLCD (2 layers): 30m-resolution coarse land cover labels from the US Geological Survey's National Land Cover Database in 15 classes (see Table 1) from two times: 2013 and 2016 (for labeling 2017 images). These labels were created in a semi-automatic way, with Landsat imagery as the principal input.

Table 1 shows the 15 classes the NLCD labels have and the four target class names we assigned to them according to [1].

Table 1 Correspondence between NLCD classes and the four target classes (cited from [7]) and assigned classes.

	NLCD class name	Label Color	Target class	Approximate class frequencies			
				W	TC	LV	I
11	Open Water		Water	98%	2%	0%	0%
21	Developed, Open Space		Low Vegetation	0%	39%	49%	12%
22	Developed, Low Intensity		Impervious	0%	31%	34%	35%
23	Developed, Medium Intensity		Impervious	1%	13%	22%	64%
24	Developed, High Intensity		Impervious	0%	3%	7%	90%
31	Barren Land (Rock/Sand/Clay)		Impervious	5%	13%	43%	40%
41	Deciduous Forest		Tree Canopy	0%	93%	5%	0%
42	Evergreen Forest		Tree Canopy	0%	95%	4%	0%
43	Mixed Forest		Tree Canopy	0%	92%	7%	0%
52	Shrub/Scrub		Tree Canopy	0%	58%	38%	4%
71	Grassland/Herbaceous		Low Vegetation	1%	23%	54%	22%
81	Pasture/Hay		Low Vegetation	0%	12%	83%	3%
82	Cultivated Crops		Low Vegetation	0%	5%	92%	1%
90	Woody Wetlands		Tree Canopy	0%	94%	5%	0%
95	Emergent Herbaceous Wetlands		Tree Canopy	8%	86%	5%	0%

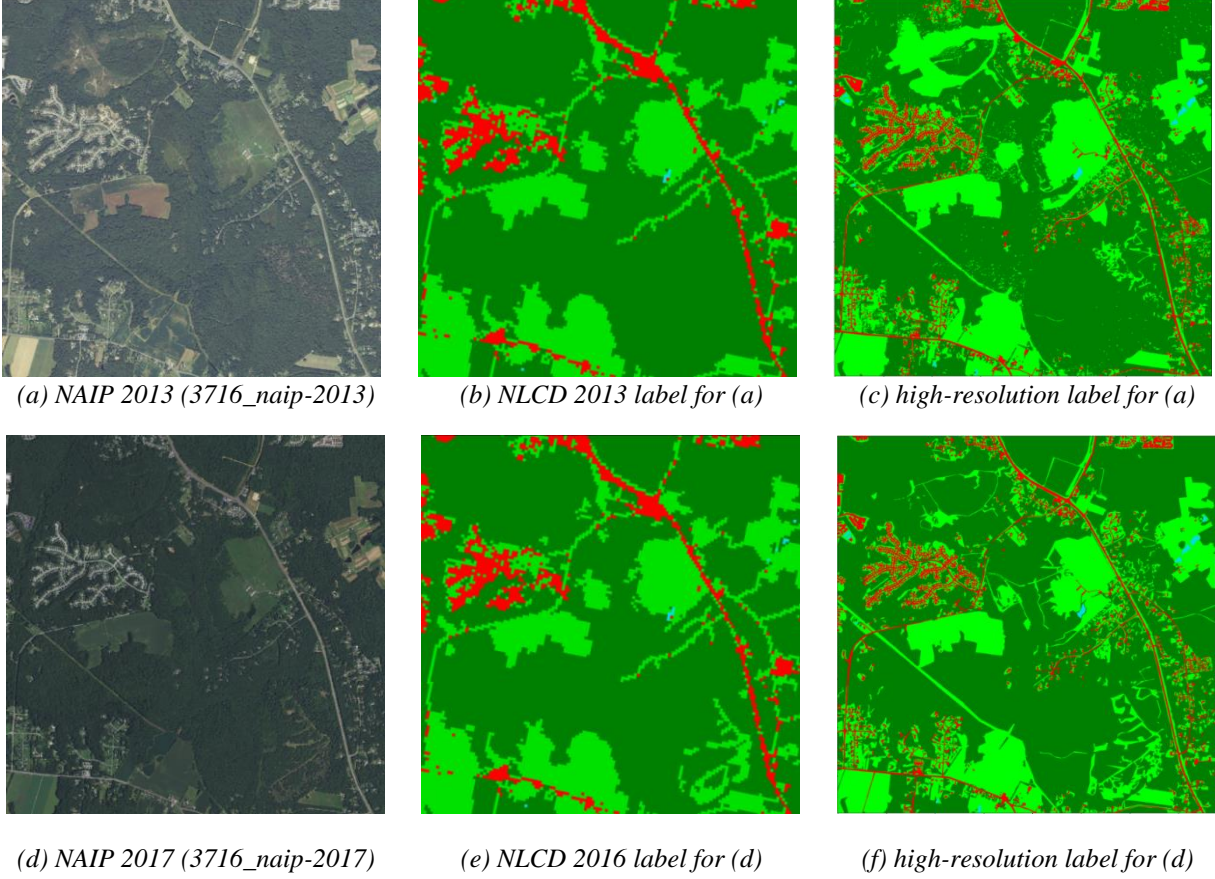
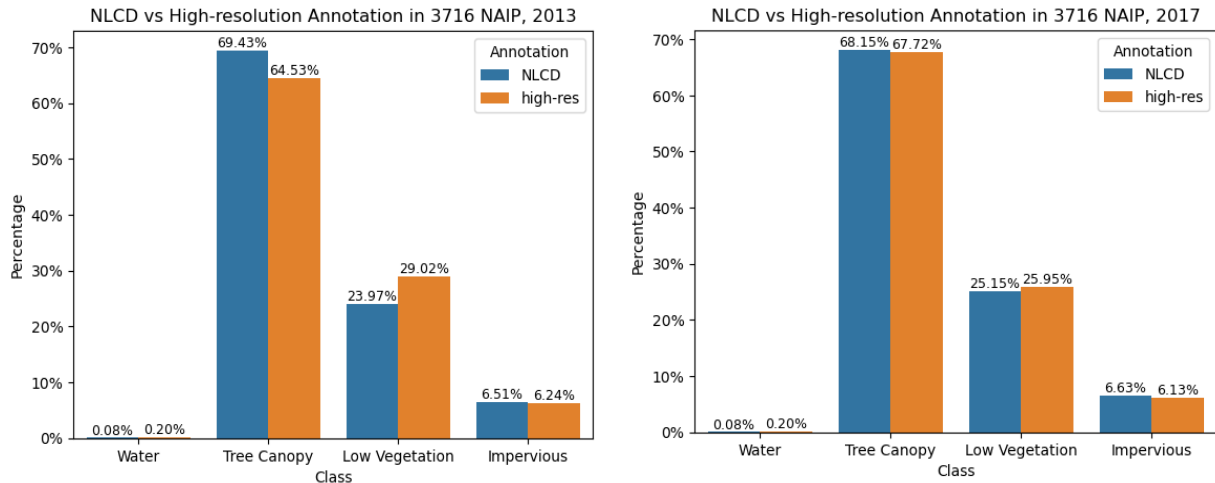


Figure 3 Sample data and their labels

Examples of the input NAIP images (3716_naip-2013 and 3716_naip-2017), their NLCD labels, and their high-resolution labels we manually annotated are shown in Figure 3. As for 3716_naip-2013, 3716 denotes the ID for the place, and 2013 shows the year when the image was shot. The most conspicuous change that happened in the period in this area would be the gain of Tree Canopy in the upper left and its loss in the bottom right. Unfortunately, NLCD labels do not capture the former difference. Also, since they are too coarse, especially concerning Low Vegetation and Impervious, the road of Low Vegetation in the middle left side is not recognized.

The training data for a year consist of 2250 NAIP aligned tiles (1m /pixel, high resolution) and NLCD (30m/pixel, low resolution). For the test data, 57 NAIP aligned tiles (1m / pixel, high resolution) selected from the 2250 training tiles are available, but their high-resolution land cover label is unavailable. On DFC 2021 contest web page on Codalab explained below, we can calculate only IoU for the test data but cannot evaluate it visually and calculate other statistics than IoU. For this reason, we classified all pixels in eight images of (width, height) = (3880, 3880) into the four target classes to create high-resolution labeling for the images using manual and semi-automatic tools in GroundWork. We also developed a code to convert a geojson file of labels on GroundWork into numpy array. This code is mathematically advanced since it converts a number of multi-polygons into an image.

For previous images, 3716_naip-2013 and 3716_naip-2017, we computed the area percentage of each label category and showed the differences between NLCD and high-resolution label images in Figure 4. Most categories for both labels have almost equal percentages. However, since high-resolution labeling is more accurate, some gaps appear. The most significant differences are the Tree Canopy and Low Vegetation percentages in 2013. These must be caused by the area and road of Low Vegetation mistaken for Tree Canopy in NLCD labels, as shown in the upper left and the middle left side in (a), (c), and (e) of Figure 3.



(a) comparison in 2013
(b) comparison in 2017
Figure 4 Area comparison between low-resolution (NLCD) and high-resolution labels

Fundamental Methodology

NLCD Difference Baseline

We studied the NLCD difference algorithm as our first baseline algorithm. It recognizes input from the 2013 and 2016 NLCD layers as the predicted target classes in 2013 and 2017 according to the target class of Table 1. They were used for each pixel, and loss and gain classes were calculated based on the change of the predicted target classes between 2013 and 2017. We used IoU scores for this method as the lower bound score to evaluate ones to neural network models.

Processing Data for Neural Network Models

After modifying the data type of the NAIP and NLCD label images and standardizing them, we added padding to them to ensure the model trained on the same size images before training because the size of images is different in some images. We trained models for the entire dataset not for the separate dataset in the years, 2013 and 2017, based on our trial results. This may give an advantage due to the much larger training data offering more potential for generalization. The model can predict the labels for each of the NAIP 2013 and NAIP 2017 images. Landcover changes can be calculated by comparing the high-resolution label images for each year in the same location.

1-Layer FCN Baseline (Multiclass Logistic Regression)

We constructed a fully convolutional network (FCN) with a single convolutional layer as our second baseline model and set its architecture as follows: the number of input channels = 4 for (Red, Green, Blue, and Near-Infrared), the number of output classes = 5 (Water, Tree Canopy, Low Vegetation, Impervious, and None), filter size = 3, stride = 1, and padding = 1. The None output class is for the case there are pixels we cannot classify into the four target classes, and we do not evaluate its IoU. Since we used cross-entropy as the loss function, the model architecture is the same as the multiclass logistic regression with 36 features (= 4 pixel colors * 9 surrounding pixels) to estimate the label of each pixel. After randomly choosing 128 pairs of NAIP images and NLCD labels out of 4500 pairs either in 2013 or 2017, we trained the model for 10 epochs in a batch size = 4 utilizing RAdam with an initial learning rate = 0.001 as an optimizer. Due to the total dataset size for this project, around 254 GB, and our computational resource limitations, we trained our model only on the part of the images. We employed this model as another baseline model to compare the results of neural network models.

Evaluation

Since we are not professionals for manual labeling, our manual labels may not be reliable. Also, labeling manually took more than 24 hours to complete only two NAIP images. Hence, we decided to use a copy environment of DFC 2021 contest web page on Codalab as a main source of evaluation. It enables us to evaluate our model performance in terms of IoUs in exactly the same way as the contest. We appreciate Professor Yokoya, one of the contest organizers, who kindly created the copy environment for us.

The inputs to Codalab are model predictions for selected 57 couples of NAIP images for 2013 and 2017. The outputs are IoU scores for loss and gain classes of each target class and the average IoU score from 2013 to 2017. Except for the pixels in the same target class in 2013 and 2017, all pixels are classified into two loss or gain classes, loss of one target class and gain of another target class. For example, if a pixel is in the Water class in 2013 and in the Impervious class in 2017, the pixel is classified into two classes, the Loss of Water class and the Gain of Impervious class. We name the loss and gain class of the four target classes, Water, Tree Canopy, Low Vegetation, and Impervious, -W, -TC, -LV, -I, +W, +TC, +LV, and +I, respectively. The average IoU score is defined as the average of the IoU scores of those eight classes. They were calculated using their ground truth labels, which professional annotators created, for some unknown (to contest participants) parts of the 57 couples of NAIP images in a particular area of Maryland in 2013 and 2017. The higher IoU scores are better.

Also, we evaluated results visually and calculate other statistics than IoU using high-resolution labels we made with manual and semi-automatic tools in GroudWork.

Baseline Model's Results

Table 2 shows the IoU scores in the eight categories of gains and losses of the four target classes and the average IoU resulting from the NLCD difference algorithm and 1-layer FCN model. Since the average IoU scores are 0.139 and 0.255, we can judge that they have only poor estimation ability. On the other hand, 1-layer FCN outperformed the NLCD difference algorithm apart from +I. So, we can consider that weakly supervised learning using neural network models

for this semantic segmentation of land cover would be basically more beneficial than using NLCD labels as they are.

Table 2 IoU scores for our baseline models. $-C$ and $+C$ denote the loss and gain of class C , respectively.

Algorithm	-W	-TC	-LV	-I	+W	+TC	+LV	+I	Avg.
NLCD diff	0.148	0.167	0.282	0.014	0.031	0.001	0.106	0.362	0.139
1-layer FCN	0.393	0.468	0.383	0.037	0.116	0.167	0.200	0.274	0.255

As can be seen in Figure 5, the 1-layer FCN model is poor at predicting most categories. This might be because of the simplicity of the model. Adding more layers to the model can create more capacity to learn and improve accuracy. In addition, the training and test images for this model were randomly selected. The more variable and diverse images would lead to better models. On the other hand, its predictions about the Impervious labels in relatively thin areas and paths are superior to NLCD labels. Therefore, ensemble learning with the shallow network can be one candidate to capture some specific areas.

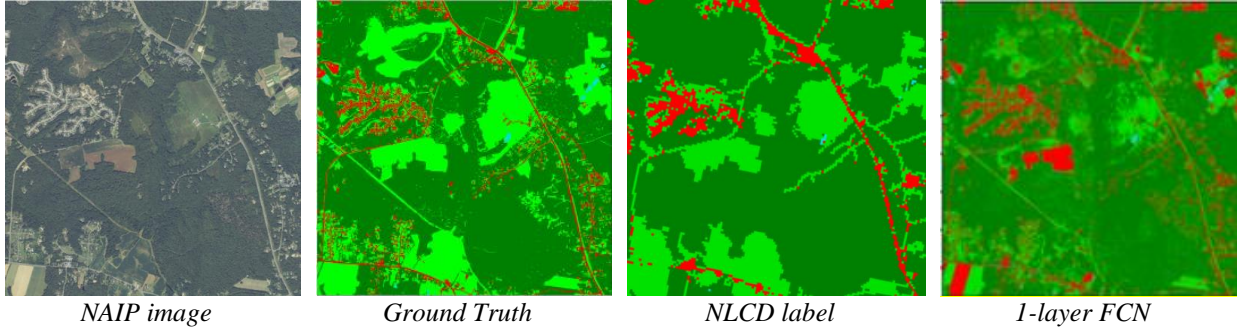


Figure 5 Comparison between NAIP image, ground truth label, NLCD label, and baseline model prediction

Literature Review

Table 3 displays the models, inputs, and outputs that have been used for land cover change labeling. We surveyed these four studies since they generated better results in compared to the other studies.

We could obtain the following considerations from Table 3 and have tried to incorporate them into our advanced model:

- It seems that multi-phase models resulted in a better result than a single-phase model
- Almost all previous studies used 5 layers consisting of 64 three-by-three with relu activation function
- To improve the result of one specific category, some studies used a specific neural network to label that category and then add and combine the result of this model to the main model
- To improve the result of the main model, extra information from different sources, such as Landsat data, can be used
- For the main model some of the famous pre-trained models such as HRNet, Deeplabv3+ were used

Table 3 Previous studies summary of models for land cover change labeling

Paper Name	Models							
[2]	Phase 1			Phase 2			Phase 3	
	Model	Input	Output	Model	Input	Output	Model	Input
	Siamese Skip_FCN (LR) (13&17)	Landsat-8	Landsat-LRL	Fusion models (separate years) (HRNet, Deeplabv3+ and Skip_FCN)	NAIP	The arithmetical average assignment is used to integrate their outputs and intersection operation is implemented to maintain the common high confidence parts in bitemporal predictions	Shadow-removal, NDVI restriction and morphological process are implemented as post-processing steps.	Intensity channel of Hue-Saturation-Intensity (HSI) color model
		NLCD with both low-resolution			NAIP-HRL (transfer 15 to 4 labels)			Near infrared (NIR) channel of images
	Siamese Skip_FCNs (HR) (13&17)	NAIP	resolution-improved labels, called "NAIP-HRL"					
the output pseudo labels of the first phase (Landsat-LRL)								
[1]	Step1			Step2			Step 3	
	Model	Input	Output	Model	Input	Output	Comparing the probability of Steps 1 and 2 assign the class	
	FCN (5 feature extraction layers and 1 classification layer)	NAIP	Pseudo-labels (4 probs for each category)	Five FCN models	Pseudo-labels	Improve the separability of water		
		NLCD			NAIP, MNDWI from 8-Landsat (New feature)			
[5]	Step1			Step1				
	Model	Input	Output	Model	Input	Output		
	FCN (5 feature extraction layers and 1 classification layer)	NAIP	Pseudo-labels (4 probs for each category)	FCN, U-Net, Deeplabv3, LinkNet, PSPNet, PAN, FPN (pixel vote)	NAIP, Pseudo-labels (4 probs for each category)	HR label resulted from both model		
		NLCD		2 class classifier just to detect water	NLCD + NLCD augmentation			
[3]	Models	Input	Output					
	FCN tile	NAIP, NLCD	HR Label					
	FCN all							
	-net all							

Our Approach

Chained Model

According to past experiences and studies, including [8], [9], and [10], we decided to adopt a chained model, the model utilizing the labels predicted by another one at the former phase as input labels. It is often used to improve the accuracy and robustness of a model by leveraging the strengths of different individual models. To choose better combination of models for the chained model and compare it with a single model, we tried the following models well-known as appropriate models for semantic segmentation tasks.

As a result of attempts to train two chained models in all possible combination patterns, the combination of U-Net18 and U-Net50 was the best from the perspective of IoU. Thus, we decided on U-Net18_U-Net50 shown in Figure 1 as our chained model. Chained models are useful when the individual models have complementary strengths and weaknesses. U-Net 18 and U-Net 50 would have different architectures with moderately deep layers compared to 1-layer FCN and 5-layer FCN. Also, since they are trained on different subsets of the data, they may have different patterns of error and may be able to capture different aspects of the data.

5-layer FCN

Due to the procedure of convolution, CNNs are much more computationally efficient than regular neural networks. They usually show higher accuracy than non-convolutional NNs, especially when there is a lot of data involved. Convolutional neural networks are often used for image classification. By recognizing valuable features, CNN can identify different objects in images. CNN can be also used in agriculture. The networks receive satellite images and can use this information to classify lands based on their cultivation level. Consequently, this data can be used for making predictions about the fertility level of the grounds or developing a strategy for the optimal use of farmland. In our project, we can use an FCN, a CNN model without Dense layers, instead of traditional CNN architecture for image classifications. FCN is suitable for the semantic segmentation task of classifying the object class for each pixel within an image, unlike traditional CNN models. We adopted a 1-layer FCN as a baseline model, but the adequately deep architectures would catch the more intricate relationship between features and target labels. To train the model, batch normalization and relu activation functions were employed, and other settings are the same as the baseline 1-layer FCN model.

U-Net

U-Net is a model proposed for semantic segmentation of medical images in 2015, which won two competitions held at medical image processing conferences in 2015. One of its main differences from FCN is that its encoder not only performs convolution processing to extract image features but also adds inverse convolution processing, upsampling, to recover the positional information of each feature in the original image. Also, concatenation is followed by regular convolution operations using skip connections. The backbone refers to the basic model structure utilized as the encoder in the convolution process. In this project, we adopted ResNet because of its excellent results in image classification. Specifically, U-Net with ResNet-18 for the backbone (hereinafter referred to as U-Net18) and U-Net with ResNet-50 (hereinafter referred to as U-Net50) were employed. We used the `segmentation_models_pytorch` library [11] with `encoder_depth = 3`, `encoder_weights = None`, and `decoder_channels = (128, 64, 64)`. Other settings are the same as the 5-layer FCN model.

Two New Ideas to be Considered

Handling the Uncertainty of the Model’s Prediction

We usually use the model’s prediction as hard labels (one of classes) not soft labels (probabilities of each class), and it is intricate to interpret and utilize original values of soft labels unless making a proper statistic. To overcome this problem, we adopted Ensemble method [4] which can show reliable uncertainty level of our model prediction to each pixel based on past studies, such as [5] and [6]. The former paper compared seven well-known modern methods. It evaluated them using common image, text, and categorical datasets, and the Ensemble method performed the best across most metrics and was more robust to dataset shift. Also, it is a model-agnostic method, and a relatively small ensemble size ($M = 5$) may be sufficient. Furthermore, the latter paper showed that the Ensemble method produced more accurate uncertainty than MC dropout which can also be used for most neural networks in the medical semantic segmentation task. As can be seen in Figure 2, we created an ensemble model with $M = 10$, which means it consists of 10 U-Net18_U-Net50 models. We trained them with 10 input datasets randomly chosen and 10 model weights randomly initialized based on [6], then calculated the entropy of soft label (predicted probabilities) normalized to be the number between 0 and 1 as the uncertainty level.

Using Dynamic World Label

In addition to NLCD labels, we also tried using Dynamic World label (henceforth DW) for a label to train our models. DW is a 10m-resolution land-cover label which is predicted automatically by a deep learning model trained on Sentinel-2 satellite images [12]. For each pixel DW has nine classes: water, trees, grass, flooded vegetation, crops, shrub & scrub, built, bare, snow & ice, and DW has two kinds of bands: label band and probability bands. Label band shows the class with the highest probability over the nine classes, and probability bands show the probability of each class, predicted by the model. Though there are already a number of global land-cover labels, as shown in Table 4, they are limited in time or of coarser resolution. On the other hand, DW is available globally and near real-time—meaning that being available until today (at least conceptually)—since June 2015 on the Google Earth Engine (GEE) and with the finest resolution (10m) among these global labels. Hence, it is of interest to see the predictive performance when using DW as a label for training. As we see later, our model trained on DW has almost the same performance as the one trained on NLCD. Moreover, since there are also local land-cover labels such as NLCD in the US and CORINE Land Cover [13] for Europe, we also investigated the performance of the model average of two models trained on NAIP+NLCD and NAIP+DW respectively. It turns out that the model average worked well as we see later.

For this study, we wrote a code to download DW for the state of Maryland. This code extracts the coordinates of the NLCD labels we have and downloads the corresponding DW. As already mentioned, DW is available from 2015, so we used only DW for 2017. Figure 6 shows one sample NAIP image (id: 3716), its manual label by authors, NLCD label, and DW label. Despite of higher resolution (10m) than NLCD (30m), DW looks rougher than NLCD. This tendency is also applicable to other NAIP images, but as we see later, the accuracy of DW is actually no worse than NLCD.

Throughout this study, we integrated the nine classes of DW (mentioned above) into four classes of interest (Water, Tree Canopy, Low Vegetation, and Impervious) as follows: “water” to Water, “trees” and “flooded vegetation” to Tree Canopy, “grass”, “crops”, “shrub & scrub”, and “bare” to Low Vegetation, and built to Impervious. There was no “snow & ice” (in the state of

Maryland for the period we specified). We downloaded DW label for each NLCD label (i.e. the same coordinates). In detail, we specified the period as June 23rd - September 29th (i.e. during the summer) in 2017 and downloaded the average probability over the period of each class for each pixel using python API of GEE. We used the probabilities as soft labels (henceforth DW 2017 soft labels). We also prepared hard labels (henceforth DW 2017 hard labels), but we did not use the label band originally prepared on GEE which is the class with the highest probability over all nine classes. Rather we took the argmax over the integrated four classes and made hard labels. Considering that DW is 10m-resolution, to save the data size, we down-sampled DW by 1/10, e.g. size 388 by 388 for a label of size 3880 by 3880 originally. We also rounded the probabilities by multiples of 0.5% and saved as uint8 upon multiplied by 200. When using in training, we transformed them back into normalized probabilities again.

Table 4 IoU List of global labels (resolution finer than 1 km) [12], [14]

Global label	time	resolution	Global label (cont.)	time	resolution
Dynamic World	2015– last day	10 m	Copernicus Global Land Service	2015– 2019	100 m
iMap 1.0	1985–2020	30 m	European Space Agency Climate Change Initiative	1992– 2018	300 m
Finer Resolution Observation and Monitoring of Global Land Cover	2010, 2015 and 2017	10 m and 30 m	NASA MCD12Q1 (MODIS Land Cover Type (MLCT) series)	2001– 2018	500 m
Global land cover	2000, 2010 and 2020	30 m			

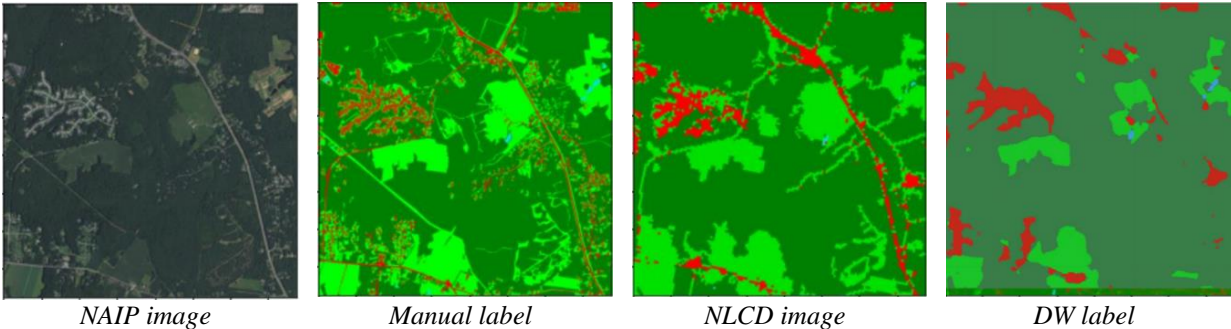


Figure 6 NAIP, manual label, NLCD, and DW label (Note that Tree Canopy is drawn with slightly different dark green color in DW for a technical reason)

Results

Our Approach

The IoU scores for baseline models and all candidate models of each class and the average IoU score (Avg.) for the 8 labels are shown in Figure 7. Note that although the total training data is large (about 270 GB), the available computational resources were limited. Accordingly, since these scores for models fitted to 128 images, about 3% of the total 4500 images, the result would be somewhat variable depending on the set of training data and initial weights of the models.

Besides, only the results of the chained model with U-Net18 and U-Net50, which was adopted based on the average IoU results, out of chained models are shown. The exact numbers can be seen in Table 5. Comparing each IoU of the NLCD difference algorithm and the other machine learning models, the machine learning models have better values for all IoU except for the baseline 1-layer FCN. Thus, the effectiveness of weakly supervised learning can be seen. In addition, U-Net18, U-Net50, and the chained model outperform the base model in all IoUs, unlike the 5-layer FCN. Since chained models that are concatenated from a shallow model to a deep model often perform well in weakly supervised learning, we tried the chained model as a candidate model in this project. As a result, it performed the best for almost all IoU scores, and the scores seemed more stable also in this modeling for land cover semantic segmentation.

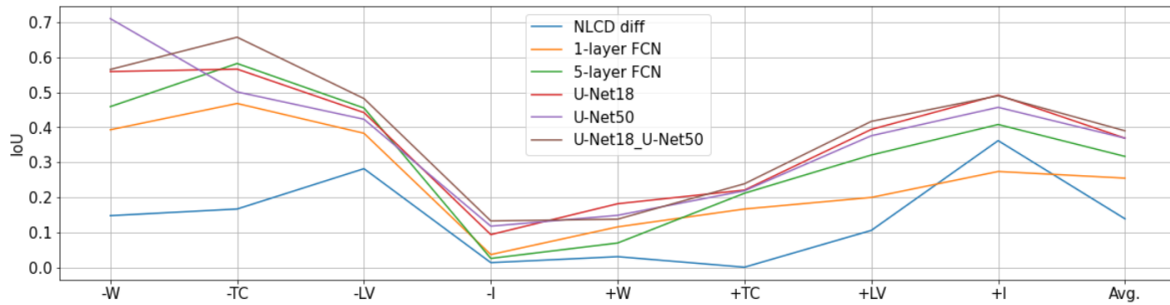


Figure 7 IoU scores for NLCD diff., baseline 1-layer FCN, and all candidate models (Avg. is the average IoU)

Table 5 The exact values of IoU scores for NLCD diff., baseline 1-layer FCN, and all candidate models

Algorithm	-W	-TC	-LV	-I	+W	+TC	+LV	+I	Avg.
NLCD diff	0.148	0.167	0.282	0.014	0.031	0.001	0.106	0.362	0.139
1-layer FCN	0.393	0.468	0.383	0.037	0.116	0.167	0.200	0.274	0.255
5-layer FCN	0.459	0.582	0.455	0.026	0.070	0.212	0.321	0.408	0.317
U-Net18	0.559	0.566	0.442	0.094	0.182	0.221	0.394	0.492	0.369
U-Net50	0.710	0.501	0.423	0.118	0.149	0.219	0.376	0.457	0.369
U-Net18_U-Net50	0.565	0.657	0.482	0.133	0.138	0.239	0.417	0.490	0.390

Figure 8 shows the high-resolution NAIP image, the corresponding ground truth label, the low-resolution NLCD label, and the estimated value by each model at one location in Maryland in 2013, 3716_naip-2013. Unlike a 1-layer FCN or 5-layer FCN, the chained model does not have the problem of sparsely mixed tree predictions among the grassland predictions. Although the results of each estimation are slightly different from those in Figure 7 because this image is only an example, the results are generally similar to the IoU scores, such as the fact that the light green Low Vegetation in the upper left corner of the correct label is not recognized at all by the NLCD label, but is captured by the U-Net18 and the chained model. Their IoU scores for loss and gain classes for Low Vegetation are also almost the highest.

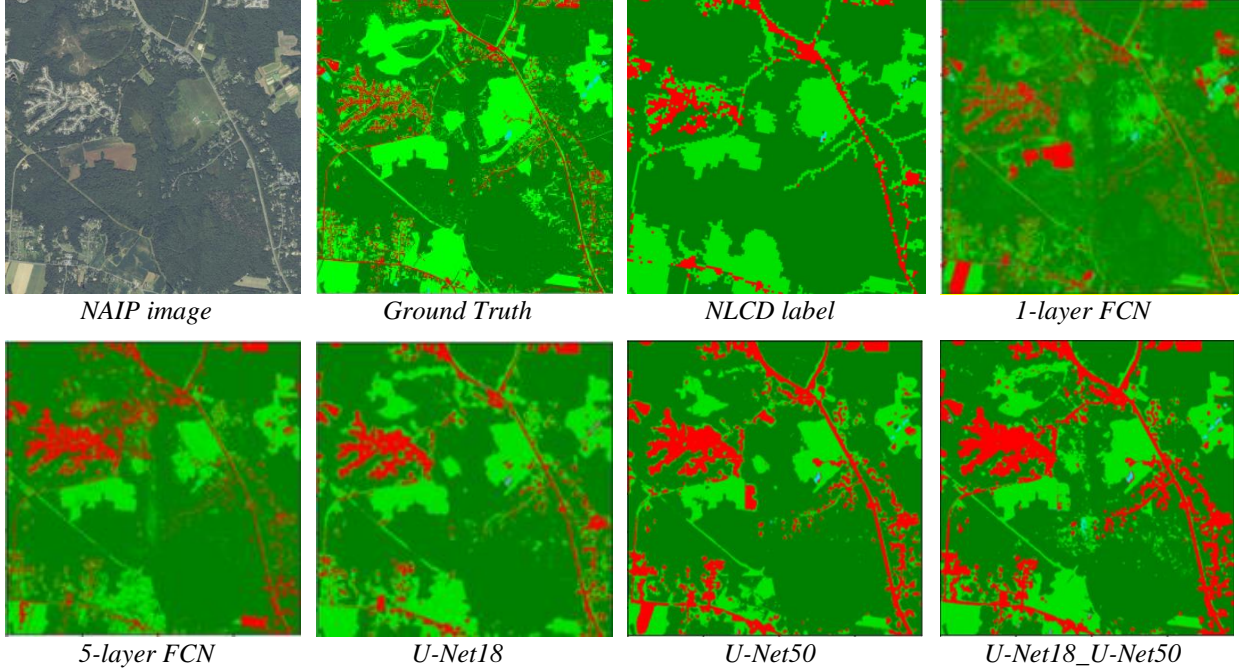


Figure 8 Comparison between NAIP image, ground truth label, NLCD label, and candidate model predictions

Handling the Uncertainty of the Model's Prediction

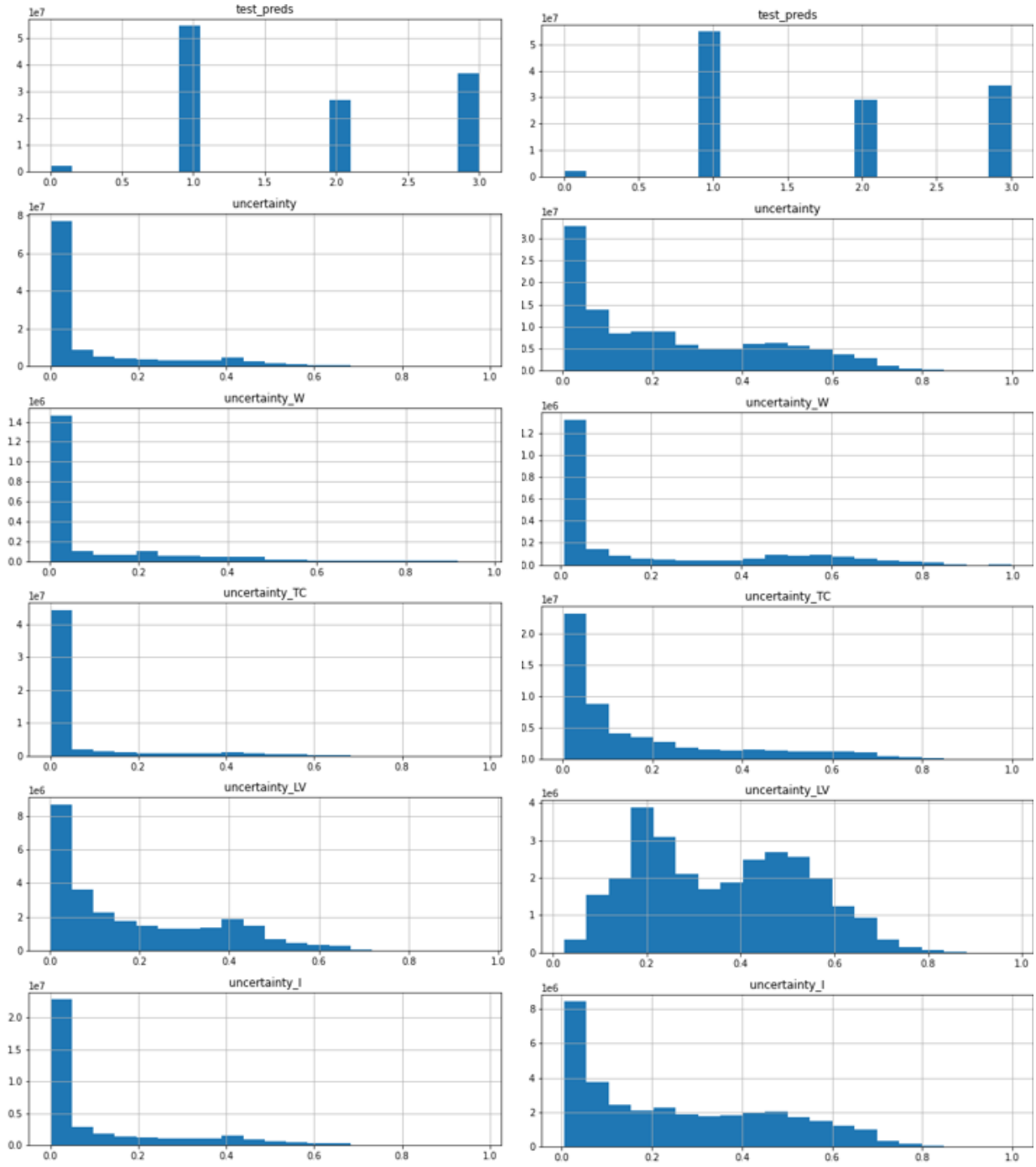
After deciding on our model as the U-Net18_U-Net50 using 128 training images, we retrained a single U-Net18_U-Net50 model and an ensemble model consisting of 10 U-Net18_U-Net50 models by 512 training images, around 11% of all 4500 training images to evaluate the accuracies of their prediction and predictive uncertainty more accurately. The IoU score of a single U-Net18_U-Net50 and ensemble model can be seen in Table 6. In the most IoU scores except for one for -I, the ensemble model outperformed the single U-Net18_U-Net50 model, including average IoU score.

Table 6 IoU scores of a single U-Net18_U-Net50 model and its Ensemble model ($M = 10$)

Algorithm	-W	-TC	-LV	-I	+W	+TC	+LV	+I	Avg.
NLCD diff	0.148	0.167	0.282	0.014	0.031	0.001	0.106	0.362	0.139
1-layer FCN	0.393	0.468	0.383	0.037	0.116	0.167	0.200	0.274	0.255
Single	0.710	0.514	0.464	0.077	0.197	0.261	0.345	0.471	0.380
Ensemble (M=10)	0.704	0.603	0.502	0.052	0.203	0.360	0.430	0.496	0.420

Figure 9 shows histograms of predictions and uncertainty levels for pixels predicted as each class of W, TC, LV, and I by the single model and the ensemble model. The histograms at the top of the graph are the ones for model predictions, where 0, 1, 2, and 3 denote Water, Tree Canopy, Low Vegetation, and Impervious class, respectively. We can see that while the ratios of model predictions for each class are almost the same, the shapes of histograms of their predicted uncertainty are quite different. All histograms for the ensemble model have fatter tails than ones for the single model. This implies that the uncertainty calculated by ensemble models reacts to the actual uncertainty level of predictions and is more reliable in land cover semantic

segmentation tasks like the results shown in other image classification and semantic segmentation tasks shown in past studies.



Histogram for the single U-Net18_U-Net50 model

Histogram for the ensemble model

Figure 9 Histogram of predictions and uncertainty for each class of W, TC, LV, and I predicted by each model

Figure 10 displays the comparison of predictions and ones with estimated uncertainty lower or higher than 80% between the single U-Net18_U-Net50 model and the ensemble model. Predictions of the ensemble model are more accurate as IoU scores show. The ensemble model's predictions with uncertainty levels lower or higher than 80% look more reasonable because its predictions with higher uncertainty are mostly in borders between lands that are used differently. It must be more complicated to assign classes to pixels on the borders than area surrounded by lands in the same usage. In addition to the implication from the histogram above, from these images, we can conclude that ensemble models with $M = 10$ would provide more accurate uncertainty levels. This method enables us to comprehend which predicted labels are more uncertain and need our manual judgment to obtain a more exact land cover gap estimation like Figure 10.

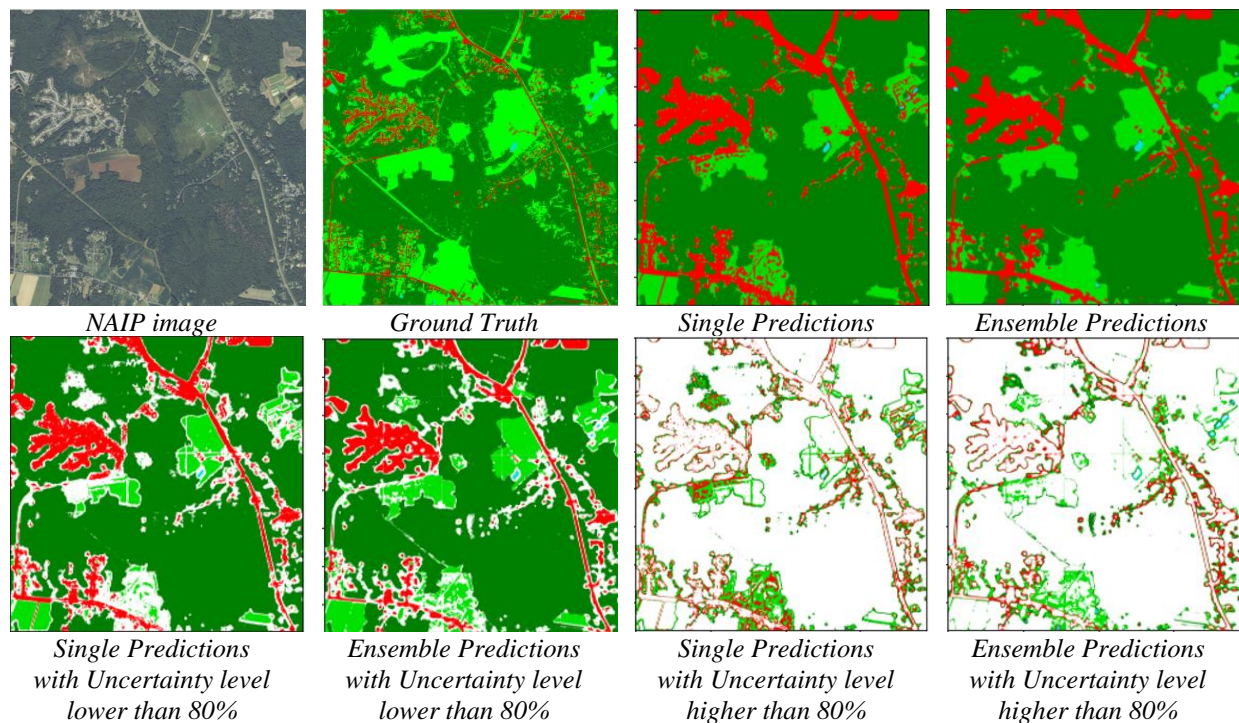


Figure 10 Comparison of predictions and ones with lower or higher predicted uncertainty between a single U-Net18_U-Net50 model and an Ensemble model

Using Dynamic World labels

We first developed an exploratory data analysis on the nature of DW. Table 7 shows a comparison of NLCD 2016 and DW 2017 over the total 2250 NAIP 2017 images for the state of Maryland. One sees that they are identical in about 80% of all pixels. The major difference is 9.9% of all pixels where NLCD is Low Vegetation and DW is Tree Canopy and 3.6% of all pixels where NLCD is Low Vegetation and DW is Impervious. So, in general, NLCD tends to say Low Vegetation while DW tends to say Tree Canopy or Impervious.

Table 7 Comparison of NLCD (2016) and Dynamic World label (2017) over total 2250 NAIP (2017) images.
Sum of the diagonal elements is 81.90%

nlcd\dw	I	LV	TC	W	sum
I	5.60%	0.90%	1.20%	0.10%	7.80%
LV	3.60%	21.30%	9.90%	0.10%	34.80%
TC	0.40%	1.00%	40.60%	0.30%	42.30%
W	0.00%	0.10%	0.50%	14.40%	15.00%
sum	9.60%	23.20%	52.20%	14.90%	100.00%

Figure 11 shows a comparison of NLCD and DW for one sample NAIP image (id = 3716). Only in this figure, Tree Canopy is colored by blue, and pixels where NLCD and DW are identical are colored by black for improving the visibility. One can see from the figure that NLCD and DW are different mainly in pixels of edges of a land cover class (e.g. Low Vegetation) or along roads.

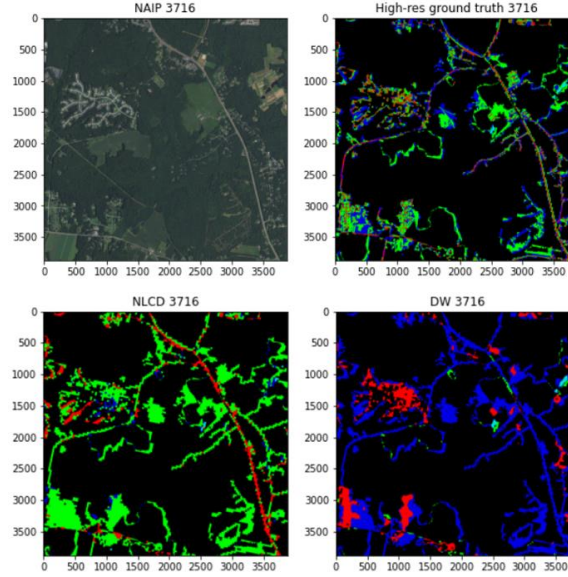


Figure 11 Comparison of NLCD (2016), Dynamic World label (2017) and our high-resolution manual label for NAIP 3716 (2017). To improve the visibility, Tree Canopy is colored by blue, and pixels where NLCD and DW are identical are colored by black

Next, we show a naïve comparison of performance between NLCD and DW using our manual labels. Specifically, we compared NLCD 2016 or DW 2017 and our high-resolution manual labels for four NAIP 2017 images and computed IoUs which are shown in Table 8. Note that these IoUs are not “loss/gain” IoUs reported in this paper. One sees that DW is by no means more inferior than NLCD in terms of IoUs. This may be at least attributed to the disadvantage of

NLCD that it is as of 2016, not 2017. This implies an advantage of DW that it is available for any years (from 2015). For ablation study, comparing with DW for 2016 can be our future task.

Table 8 IoU comparison of NLCD 2016 and DW 2017 for four manual ground-truth labels by authors

Class	NLCD	DW
W	71.7%	74.3%
TC	65.6%	74.1%
LV	51.5%	49.0%
I	38.0%	42.1%
simple ave	56.7%	59.9%

Finally, we investigated the performance of models trained on NAIP 2017 images and DW 2017 labels, compared with NAIP 2017 images and NLCD 2016 labels. Specifically, we compared the following four models:

1. Model trained on NLCD 2016
2. Model trained on DW 2017 hard labels
3. Model trained on DW 2017 soft labels
4. Model average of 1. and 3. (taking average of outputted probability of each model)

Since there are no DW labels for 2013, we trained our models from 1. to 3. using either NLCD 2016 or DW 2017, and apply those models not only for NAIP 2017 images but also NAIP 2013 images as well. For models, we only tried the same 5-layer FCN as already introduced but without batch normalization, i.e. the same structure as used in the baseline model GitHub [15]. We used randomly selected 2000 pairs of NAIP 2017 images, NLCD 2016 and DW 2017 labels, and we trained our models for 0.5 epoch (i.e. 1000 images). Considering the difference in the brightness between NAIP 2013 (brighter) and 2017 (darker), we applied data augmentation with respect to the brightness of the NAIP image which is explained in more detail in the Appendix. So, the each pair of NAIP image and NLCD or DW label will be augmented to 2 pairs (original NAIP image and its brighter one with its label unchanged). Hence, the experiment flow is as Figure 12.

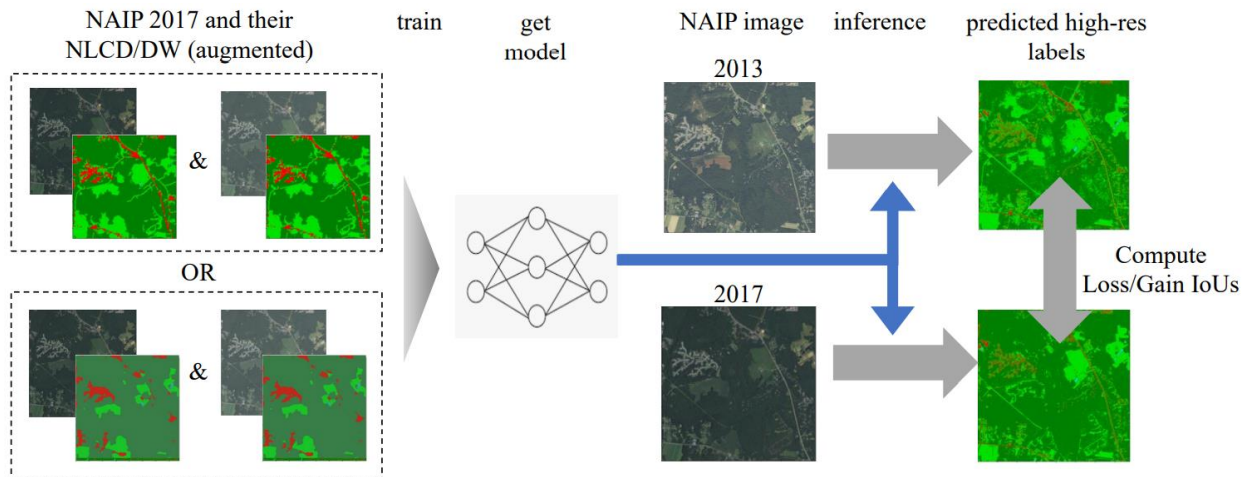


Figure 12 Experiment flow

The results are summarized in Table 9. First, DW soft label (solid orange) has similar performance to NLCD. This is important result since it implies the possibility that we can use DW instead of NLCD without degrading the predictive performance at least for the state of Maryland. Second, DW soft label is slightly better than DW hard label in average and for many categories, which encourage the use of soft labels rather than hard labels when they are available. Finally, the model average of NLCD and DW soft label performed better than both in average and for almost all categories, which suggests one way to combine global and local labels. Note that its average IoU was 0.3949, which is comparable to the 5-layer FCN models trained on both NAIP+NLCD 2013 and 2017.

Table 9 Result of IoU scores of 5-layer FCN baseline models

	Ave	W-	TC-	LV-	I-	W+	TC+	LV+	I+
NLCD	0.3781	0.3480	0.4954	0.5397	0.3360	0.1253	0.2604	0.4992	0.4208
DW (hard)	0.3557	0.3190	0.4767	0.5446	0.2999	0.1106	0.1879	0.4688	0.4381
DW (soft)	0.3727	0.4203	0.4906	0.5226	0.2304	0.1474	0.2449	0.4476	0.4776
Model average: NLCD + DW(soft)	0.3949	0.3840	0.4973	0.5683	0.3361	0.1380	0.2496	0.5071	0.4789

Conclusion

We implemented an ensemble model of the 10 chained model with U-Net18 and U-Net50 to obtain more accurate predictions and reliable uncertainty level of predictions, using the Ensemble method. The IoU score for the ensemble model was the best in all the candidate models, a single 1-layer FCN, 5-layer FCN, U-Net18, U-Net50, and all possible chained models using them. Moreover, we could confirm that the Ensemble method is appropriate not only for image classification and medical semantic segmentation tasks but also for this land cover semantic segmentation. Furthermore, 10 models are sufficient to make reliable ensemble models in this task.

On Dynamic World labels (DW), DW has comparable performance as a weak label to NLCD. This implies that we might not have to use NLCD anymore. Also, using probabilities rather than deterministic labels leads to slightly better performance for DW. This encourages using soft labels when they are available. Moreover, the model average of NLCD and DW gives slightly better performance than either NLCD or DW only, which suggests a possible way of combining local and global labels to improve performance.

Next Steps

For the next step, investigating more models and using some other image segmentation can be implemented to see if they can improve the result. We can utilize findings about the Dynamic World label to improve the result of final models. Also, other method to show uncertainty, such as Deterministic Uncertainty Quantification (DUQ) would not only make accuracy of models' predictive uncertainty better but also reduce not a few computational cost that the Ensemble method requires.

Ethical considerations

For this project there is no ethical consideration. The dataset is public and available for all.

Contribution

- Ashkan Bozorgzad: Data Cleaning, Final report, Developed the initial code of baseline model, and Prepared poster and presentation slides
- Hari Prasad: Exploratory Data Analytics, Research about uncertainty of models - ensemble and DUQ, Prepared poster and final report.
- Karveandhan Palanisamy: Exploratory Data Analysis, Comparison of NLCD and Dynamic World label. Researched into alternative of dynamic world labels: LoveDA and Dynamic EarthNet, poster preparation, and final report.
- Masataka Koga: Team Captain: Set up logistics, planned schedule, and managed progress. EDA. Adopt the Ensemble method to show uncertainty, implemented, and evaluated it. Trained and evaluated baseline models, candidate models, chained models, and ensemble models. Reviewed, made, and finalized first progress report, poster, abstract, and final report. Created, refactorized, and finalized all codes mainly other than ones for DW label analysis. Cleaned and preprocessed data, and created high-resolution labels for test data
- Yewen Zhou: Made heatmaps and bar plots comparing distributions of classes in high-res and NLCD images, Developed code for soft labeling using Tensorflow Keras, Researched ways to set up VMs with T4 and A-100 GPUs on GCP and recorded videos, and Organized GitHub repo
- Yuki Ikeda: Developed the initial code of baseline model using PyTorch, developed analysis on Dynamic World label (downloading, EDA, training, writing this report and making slides), established the way of reliable evaluation using DFC2021 contest page on Codalab (including developing the initial code for submission), explored and shared an efficient way to use GroundWork for manual labeling, and developed the code to convert a geojson file of a manual label on GroundWork into numpy array

References

- [1] K. Malkin, C. Robinson and N. Jojic, "High-resolution land cover change from low-resolution labels: Simple baselines for the 2021 IEEE GRSS Data Fusion Contest," arXiv preprint arXiv:2101.01154, 2021.
- [2] Z. Li, F. Lu, H. Zhang, G. Yang and L. Zhang, "Change cross-detection based on label improvements and multi-model fusion for multi-temporal remote sensing images," *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, pp. 2054-2057, 2021.
- [3] L. Tu, J. Li and X. Huang, "High-resolution land cover change detection using low-resolution labels via a semi-supervised deep learning approach - 2021 IEEE data fusion contest track MSD," *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, pp. 2058-2061, 2021.
- [4] B. Lakshminarayanan, A. Pritzel and C. Blundell, "Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles," in *the 31st International Conference on Neural Information Processing Systems*, 2017.
- [5] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. V. Dillon, B. Lakshminarayanan and J. Snoek, "Can You Trust Your Model's Uncertainty? Evaluating

- Predictive Uncertainty Under Dataset Shift," in *33rd Conference on Neural Information Processing Systems*, 2019.
- [6] A. Mehrtash, W. M. Wells III, C. M. Tempany, P. Abolmaesumi and T. Kapur, "Confidence Calibration and Predictive Uncertainty Estimation for Deep Medical Image Segmentation," *IEEE Transactions on Medical Imaging*, vol. 39, no. 12, pp. 3868-3878, 2020.
 - [7] "2021 IEEE GRSS Data Fusion Contest: Track MSD," [Online]. Available: <https://www.grss-ieee.org/community/technical-committees/2021-ieee-grss-data-fusion-contest-track-msd/>.
 - [8] Q. Bao, Y. Liu, Z. Zhang, D. Chen, Y. Yang, L. Jiao and F. Liu, "MRTA: Multi-resolution training algorithm for multitemporal semantic change detection," *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, pp. 2062-2065, 2021.
 - [9] Z. Li, F. Lu, H. Zhang, L. Tu, J. Li, X. Huang, C. Robinson, N. Malkin, N. Jojic, P. Ghamisi, R. Hänsch and N. Yokoya, "The outcome of the 2021 IEEE GRSS Data Fusion Contest-Track MSD: multitemporal semantic change detection," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 1643-1655, 2022.
 - [10] Z. Zheng, Y. Liu, S. Tian, J. Wang, A. Ma and Y. Zhong, "Weakly supervised semantic change detection via label refinement framework," *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, pp. 2066-2069, 2021.
 - [11] P. Iakubovskii, "Segmentation Models Pytorch," 2019. [Online]. Available: https://github.com/qubvel/segmentation_models.pytorch.
 - [12] C. F. Brown, S. P. Brumby, B. Guzder-Williams, T. Birch, S. B. Hyde, W. Czerwinski, V. J. Pasquarella, R. Haertel, S. Ilyushchenko, K. Schwehr, M. Weisse, F. Stolle, C. Hanson, O. Guinan, R. Moore and A. M. Tait, "Dynamic World, Near real-time global 10 m land use land cover mapping," *Sci Data*, vol. 9, no. 251, 2022.
 - [13] D. García-Álvarez, J. L. Hinojosa, F. J. J. Pérez and J. Q. Villaraso, "General Land Use Cover Datasets for Europe," Springer, Cham, 2022, p. 313–345.
 - [14] H. Liu, P. Gong, J. Wang, X. Wang, G. Ning and B. Xu, "Production of global daily seamless data cubes and quantification of global land cover change from 1985 to 2020 - iMap World 1.0," *Remote Sensing of Environment*, vol. 258, p. 112364, 2021.
 - [15] K. Malkin, C. Robinson and N. Jojic, "High-resolution land cover change from low-resolution labels: Simple baselines for the 2021 IEEE GRSS Data Fusion Contest," 2021. [Online]. Available: <https://github.com/calebrob6/dfc2021-msd-baseline>.

Appendices

A. EDA

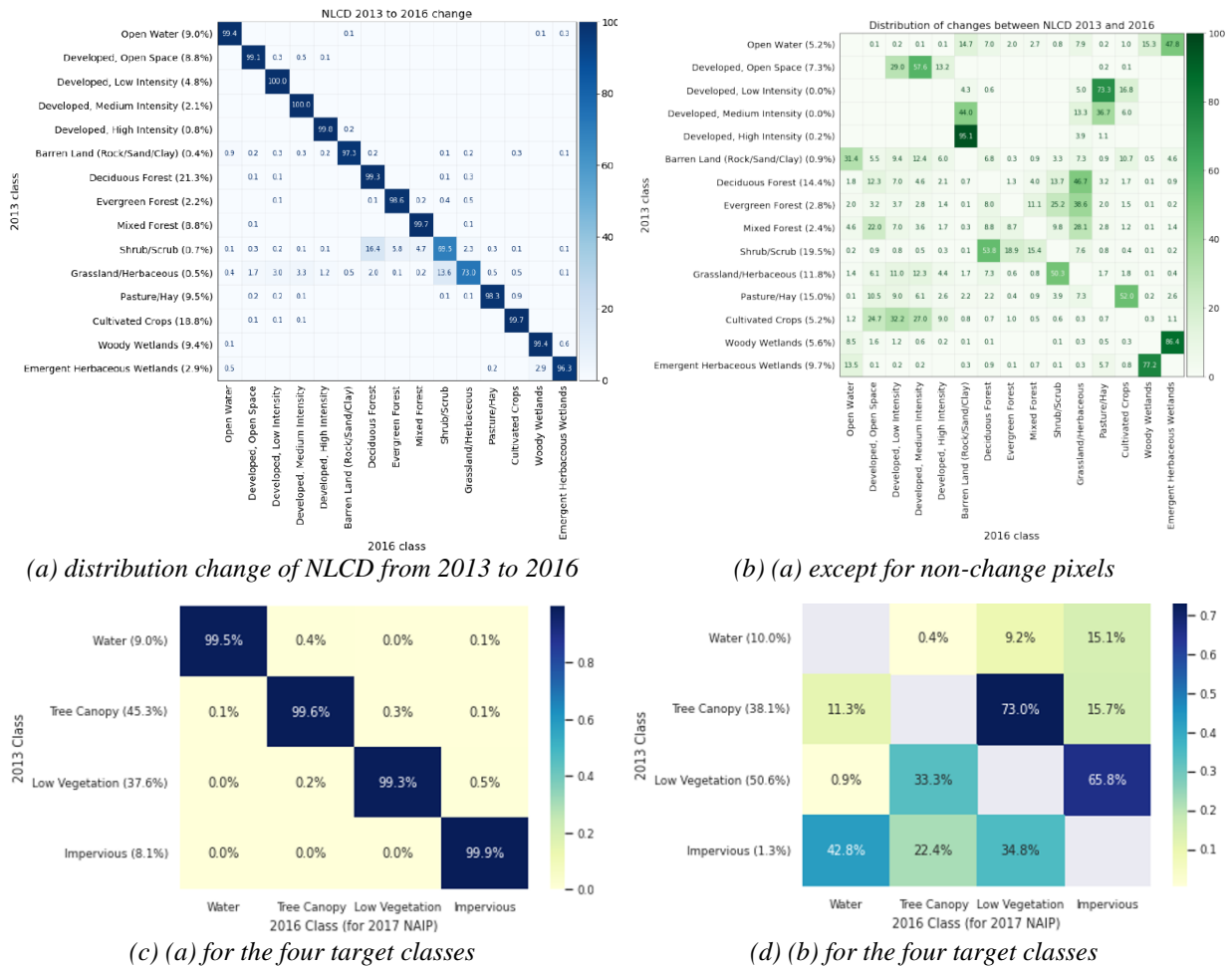


Figure 13 Distribution of changes between NLCD 2013 and 2016

In Figure 13, (a) and (b) were cited from [1], and (c) and (d) were calculated by us using the numbers in (a) and (b). In graphs (a) and (b) in Figure 13 cited from [1], the distribution of the 15 classes in the NLCD labels in 2013 (left axis) and its change between 2013 and 2016 can be seen. The numbers of elements in the matrices denote the distribution of 2016 classes for pixels of each 2013 class in percentage. (b) shows the distributions for the pixels limited to those where their labels changed between 2013 and 2016. From graph (a), we can see that Deciduous Forest (21.3%) and Cultivated Crops (18.8%) accounted for the broadest part of Maryland in 2013, while Developed, High Intensity (0.8%), Shrub/Scrub (0.7%), Grassland/Herbaceous (0.5%), and Barren Land (0.4%) the minor part. Moreover, 30.5% of Shrub/Scrub and 27.0% of Grassland/Herbaceous in 2013 became other classes by 2017. In graph (b), 60% of changes was composed of Shrub/Scrub (19.5%), Pasture/Hay (15.0%), Deciduous Forest (14.4%), and Grassland/Herbaceous (11.8%) in 2013. Also, some changes, such as Developed, High Intensity to Barren Land (95.1%) and Woody Wetlands to Emergent Herbaceous Wetlands (86.4%), seem more common than others.

The distribution changes for the four target classes can be seen in graphs (c) and (d). The distribution in 2013 (left axis) in (c) shows that most part in Maryland is Tree Canopy (45.3%) or Low Vegetation (37.6%), and there was little change in land cover during the time because more than 99% of each class remained the same category. As for graph (d), the change of labels from Impervious (1.3%) in 2013 appears the least. Losses of Tree Canopy into Low Vegetation (73.0%) and those of Low Vegetation into Impervious (50.6%) are distinctive points. These graphs imply that there would be some patterns of land cover changes, and we can capture them by moderately deep neural networks. Also, it may be beneficial to adopt weighted loss in training them because training data of some classes and changes are more scarce than others.

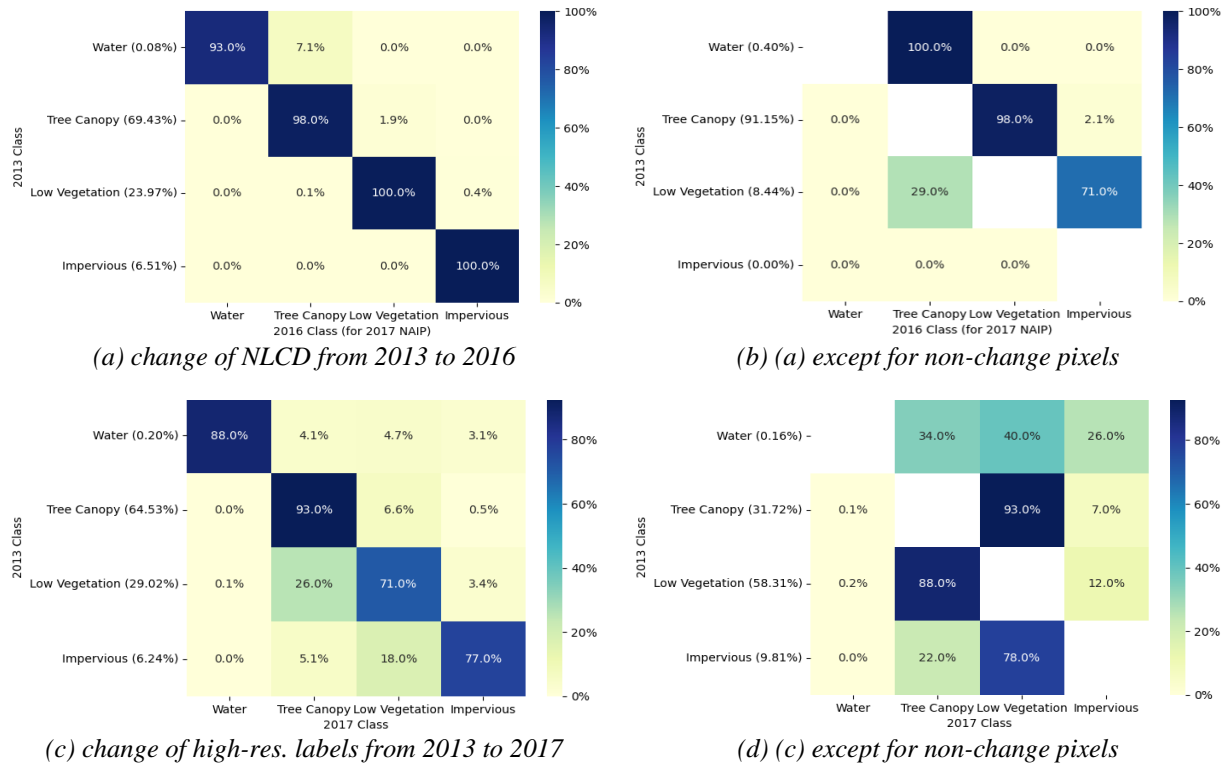


Figure 14 Distribution change Comparison of NLCD and high-res. label for 3716_naip-2013 and 3716_naip-2017

Figure 14 shows distribution changes of NLCD and high-resolution labels for 3716_naip-2013 and 3716_naip-2017. From graphs (a) and (c), we can see that more Low Vegetation (29.0%) and Impervious (23.0%) were lost between 2013 and 2017 than the volume NLCD implied. Then, the most significant loss in 2013 was not Tree Canopy (91.15%) as (b) indicates, but Low Vegetation (58.31%) shown in (d). Most losses in Low Vegetation led to gains in Tree Canopy (88.0%). We can guess that the unrecognized area and road of Low Vegetation in NLCD in 2013 mentioned above generated these gaps. Furthermore, the change volume of Impervious was not 0.00% but 9.81% and was removed or covered by Low Vegetation mainly (78.0%) in 2017. Water was filled by Tree Canopy, Low Vegetation, and Impervious in nearly equal proportion. These differences would be a pile of minor ones due to the roughness of NLCD labels.

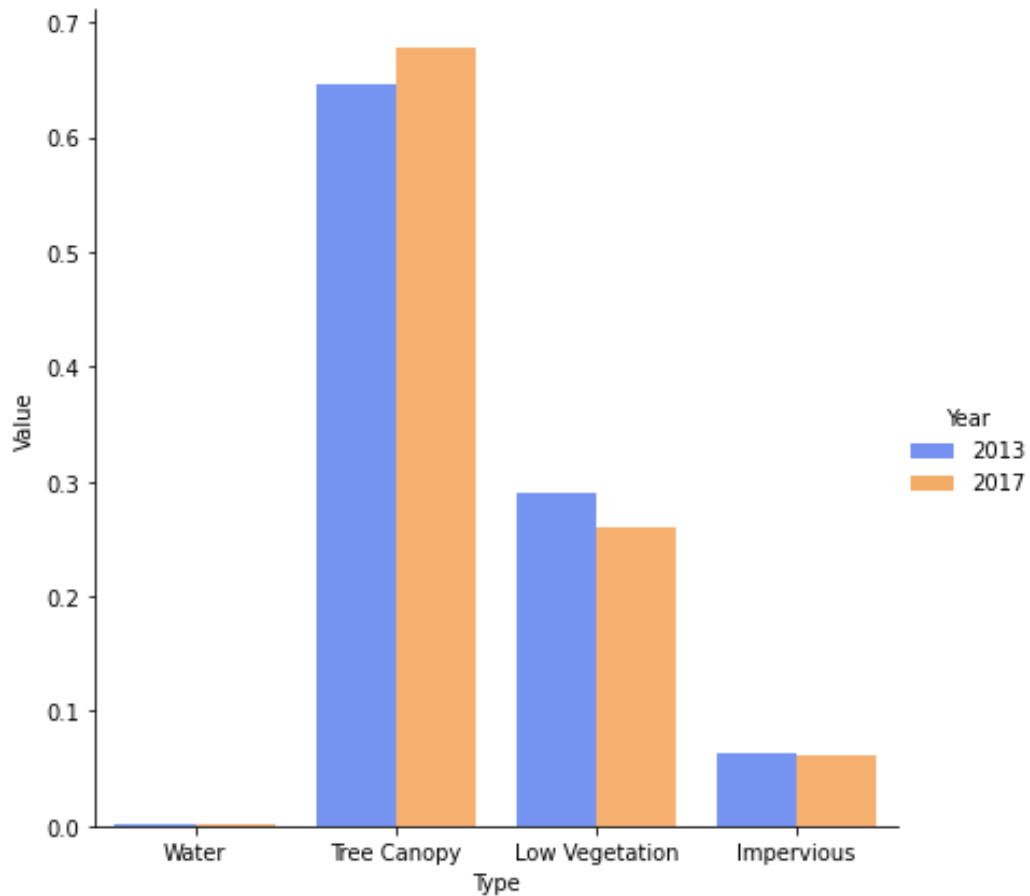


Figure 15 Area comparison in high-resolution labels between 2013 and 2017

Figure 15 shows the area comparison in the high-resolution labels in 2013 and 2017 in a particular zone, Maryland state (3716_naip-2013 and 3716_naip-2017). We can see an increase in the proportion of Tree Canopy and a decrease in Low Vegetation. The proportion of Water has been little and almost remains the same.

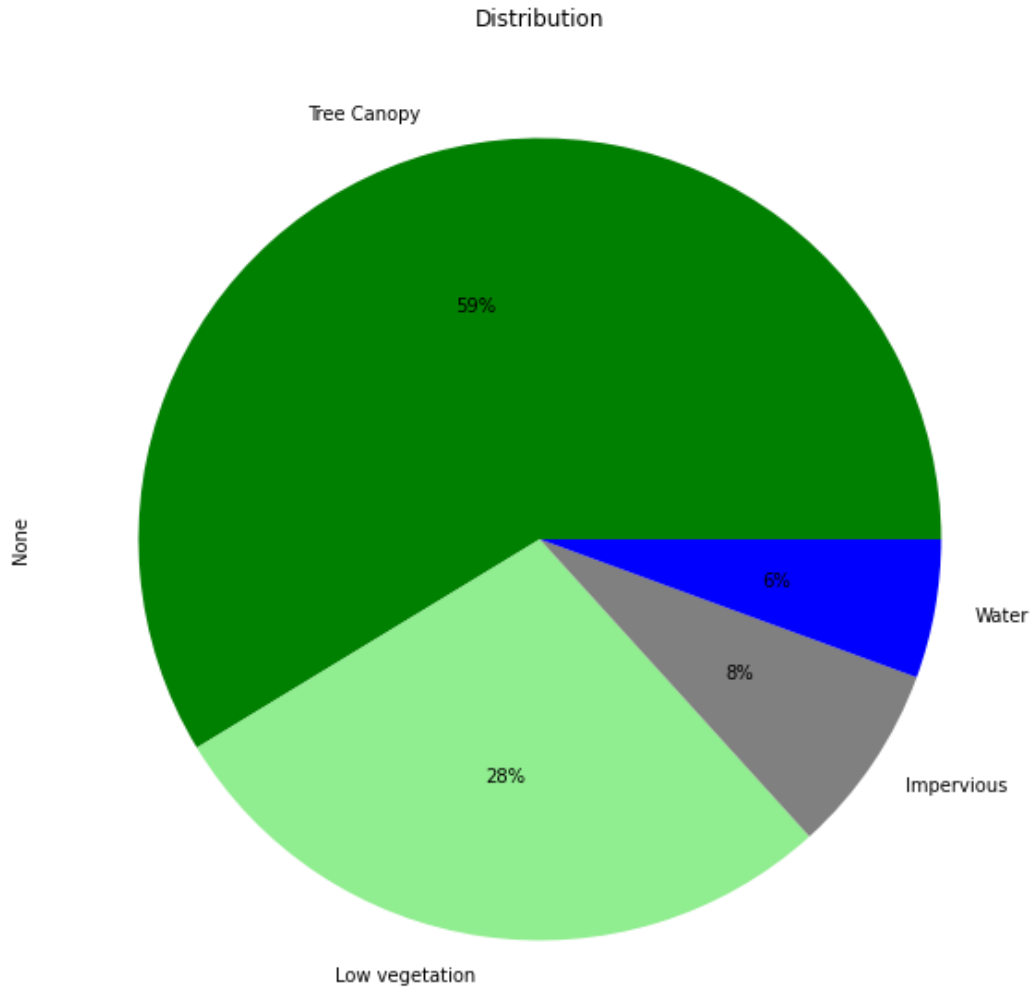


Figure 16 The distribution of the four target classes in an area other than 3716 NAIP

Figure 16 shows the distribution of the four target labels in an area of Maryland different from 3716 NAIP. This area includes the four target classes in a ratio similar to the average in (c), Figure 13 (in 2013, Water: 9.0%, Tree Canopy: 45.3%, Low Vegetation: 37.6%, and Impervious: 8.1%), in contrast to 3716 NAIP. Therefore, we can employ this area to tune the general purpose model and the other with distinctive distribution, like the one for 3716 NAIP, to sophisticate the model that specializes in discerning a specific type of area.

B. Data augmentation with respect to brightness

When we trained models on NAIP 2017 and either DW 2017 or NLCD 2016, we applied data augmentation with respect to brightness of NAIP 2017 images. Specifically, we prepared one brighter transformed image for every NAIP 2017 image that we used for training with its label (either DW 2017 or NLCD 2016) unchanged. Since NAIP 2013 images are generally brighter than their corresponding NAIP 2017 ones, this is to generalize the model even to brighter NAIP 2013 images. In more detail, we applied γ -transformation to the numbers of the RGB channels and the infrared channel of each pixel of NAIP 2017 images by the following (Equation 2)

$$x_{transformed} = 255 \left(\frac{x}{255} \right)^{1/\gamma}, \quad (\text{Equation 2})$$

where γ is set to 1.6 and x is the number of each channel. γ -transformation uses the exponential relationship between the input light volume and the output signal intensity by image sensors, and it is popular in image analysis to adjust brightness. In our case, this will increase the numbers of each channel and make the original images brighter. Note that we normalized all input images by dividing by 255, but we did not standardize them, i.e. we did not subtract mean or divide by standard deviation. The figure below shows the flow of the data augmentation in the case of NLCD 2016 label.

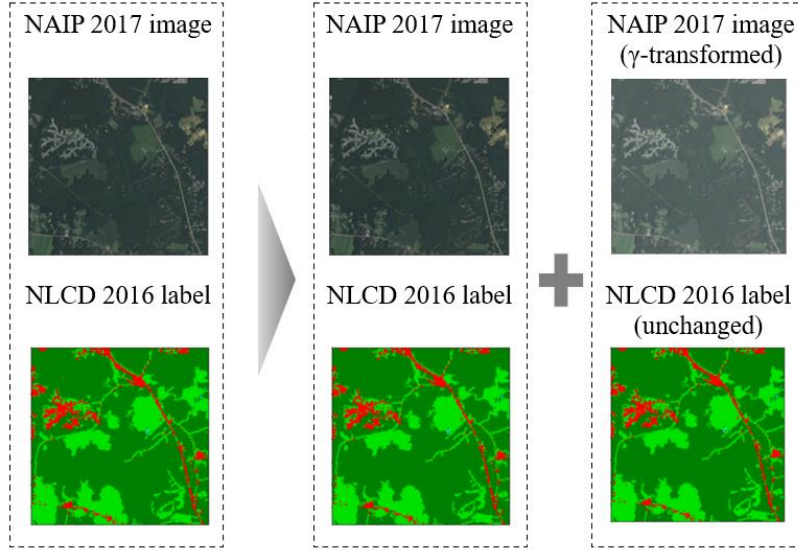


Figure 17 Data augmentation with respect to brightness in case of NLCD label

The effect of this data augmentation is remarkable. Figure 18 shows the predictions of a model trained with and without data augmentation (trained NAIP 2017 and NLCD 2016). Without data augmentation, the model trained on NAIP 2017 did not generalize well to NAIP 2013. However, training on augmented dataset led to a model generalized even to NAIP 2013.

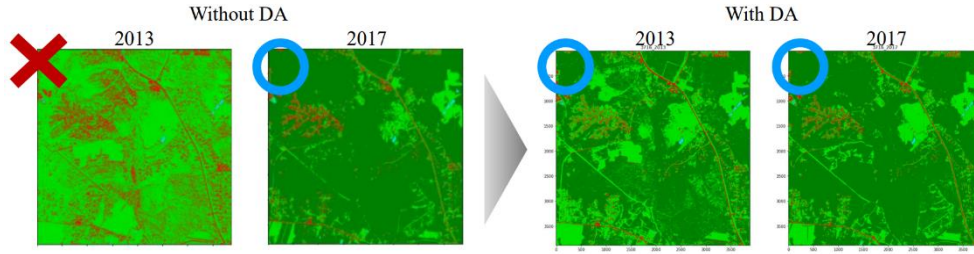


Figure 18 Prediction of models with and without data augmentation

Note that another possible way is standardization of input images, however, if images for training and images for which predictions are made are quite different (e.g. taken from quite different regions), the standardized distributions of numbers of each channel would be quite different, which can still cause failures in model predictions. Also, it is not feasible for online prediction where we cannot know the distribution of the input images for which predictions are made.