Land Cover Classification at the Wildland Urban Interface using High-Resolution Satellite Imagery and Deep Learning

Mai H. Nguyen*, Jessica Block‡, Daniel Crawl*, Vincent Siu†, Akshit Bhatnagar†, Federico Rodriguez†, Alison Kwan†, Namrita Baru*, Ilkay Altintas*

Abstract—**Land cover classification analysis from satellite imagery is important for monitoring change in ecosystems and urban growth over time. However, the land cover classifications that are widely available in the United States are generated at a low spatial and temporal resolution, so that the spatial distribution between vegetation and urban areas in the wildland urban interface is difficult to measure. High spatial and temporal resolution analysis is essential for understanding and managing changing environments in these regions. This paper describes an end to end satellite data ingestion and analysis pipeline using deep learning on high resolution satellite imagery for generating pixel-based land cover classification.**

Index Terms—**satellite image analysis, deep learning, land data products, CNNs, U-Nets**

I. Introduction

Understanding land cover classification has applications in many areas including planning for fire hazards and response. Understanding land cover at the wildland urban interface (WUI) is especially important because the proximity of structures to different vegetation significantly affects how a fire will burn in that area. There are many data land cover products that classify vegetation in the United States and others that classify urban (or impervious) surfaces, but none investigate vegetation at the urban interface with the intention to understand how the WUI is changing.

WIFIRE [1], [2] is an integrated system for wildfire analysis, which integrates networked observations such as heterogeneous satellite data and real-time remote sensing data, with computational techniques in signal processing, visualization, modeling, and data assimilation to provide a scalable method to monitor such phenomena as weather patterns that can help predict a wildfire's rate of spread. The amount, size and moisture content of surface fuels determine how fast a fire spreads, how hot it burns and how high its flames reach. Accurate and up-to-date fuel maps are critical for accurately modeling wildfire rate of spread and potential burn areas. Our goal is to create a machine learning method for automating a burnability or "fuel" model from satellite imagery. However, in order to do that, we must first refine a technique for land cover classification. Land cover determines a type of vegetation or density of urban land. Fuel is determined from additional attributes that cannot necessarily be determined from satellites such as canopy height, percent of live versus dead vegetation, and the burn history of that area. Therefore our goal for this paper is to create accurate land cover as a step towards identifying fuels.

New advances in satellite imagery now provide data at an unprecedented rate and resolution. This remote sensing data combined with data science techniques provide a unique opportunity to monitor natural resource and development activities on an ongoing basis.

Deep learning on satellite imagery represents a number of typical big data challenges related to the data volume and computational scale required to analyze and interpret. Deep learning models are composed of many layers of interconnected processing units, allowing for the learning of data representations at multiple and increasingly complex and task-specific levels of abstraction, leading to automatic feature learning for prediction tasks across many domains, including image analysis. In this paper, we present a data-driven approach to measure the WUI using deep learning on satellite imagery at high spatial and temporal resolution for a better understanding of fire hazard and response needs.

Based on this motivation and our prior work [3], the new contributions of this paper are as follows:

(i) An end to end satellite data ingestion and analysis pipeline using deep learning;

(ii) Land cover maps generated by the presented analytical pipeline that defines vegetation and urban areas at the WUI on a pixel by pixel basis; and

*(*iii) A comparison of the experimental results from using convolutional neural network (CNN) and U-Net models for detecting land cover classes.

The rest of this paper is organized as follows: Section II reviews related work, Section III discusses our data and methods, and Section IV describes our results. Finally, we conclude and discuss future work in Section V.

II. Related Work

A. Vegetation Maps using Satellite Imagery

There are several existing map products derived from satellite imagery made available for analyzing fire. These free products

Corresponding Authors: Mai Nguyen (mhnguyen@ucsd.edu) and Ilkay Altintas (ialtintas@ucsd.edu)

*San Diego Supercomputer Center, University of California, San Diego

‡Qualcomm Institute, University of California, San Diego

†Master of Advanced Studies in Data Science and Engineering University of California, San Diego

978-1-5386-5035-6/18/$31.00 ©2018 IEEE

are derived from Landsat [4] satellite imagery, which are 30m/pixel resolution.

LANDFIRE [5], the Landscape Fire and Resource Management Planning Tools, provides vegetation and fire fuel data for resource planning and analysis every two years by using a workflow that combines Landsat imagery, vegetation plots throughout the United States, and ecological expertise to create the seamless product across the US. The result is a map that describes ecosystem-scale vegetation types and urban areas.

The North American Forest Dynamics (NAFD) [6], managed by NASA, is intended to show forest disturbance over the conterminous United States using Landsat data between 1986-2010. This product provides annual maps for these dates that describe whether pixel is water, forest, not forest, and how much that pixel has been disturbed from the previous year.

National Land Cover Database (NLCD) [7], housed by the Multi-Resolution Land Characteristics Consortium, classifies the conterminous US into 16 general land cover classifications into general vegetation types and urban densities, for the purpose of measuring change from 2001-2011.

Forest Inventory Analysis (FIA) [8], managed by the US Forest Service, creates maps that quantify forest disturbance for carbon accounting across the conterminous United States.

The Hansen Global Maps of Forest Cover Change [9] quantifies globally the growth and loss of forest cover between 2000-2012 using the help of the Google Earth Engine.

While all of these efforts focus on forests and forest health with a specific focus on forest disturbance, they do not address the diversity and challenges of vegetation or fuels at the edges or at some times, insides, of urban areas.

B. Deep Learning for Image Analysis

Convolutional neural networks (CNNs), a particular type of deep learning model, are extensively used and are now the de facto approach for several image tasks such as image classification and object detection, as can be seen from the outcomes of the highly influential ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [10]. CNN performance on the ImageNet database [11] has surpassed human performance on a standard image classification task [12].

Several groups have applied CNNs to land cover classification. For example, there are studies to classify crop types using medium-resolution satellite imagery [13], analyze urban land use patterns from aerial images [14], classify land cover from geo-tagged field photos [15], and study urban environment patterns from satellite images [16].

Another type of deep learning model known as U-Net have shown to be very effective at image segmentation in biomedical and medical applications [17]. Recently, U-Nets have also been applied to other domains, such as scene segmentation (e.g., [18]) and satellite image analysis (e.g., [19]).

Our work applies deep learning to satellite images to study land cover classification. Our analysis pipeline makes use of up-to-date, high-resolution satellite imagery, and classifies land cover types at the pixel level rather than the image level in order to provide the granularity necessary to distinguish between urban materials from vegetation, and where they intermix.

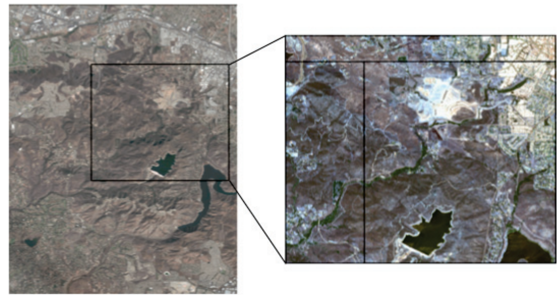

Fig. 1: Study area from Escondido, California.

III. Approach

A. Data Acquisition and Preprocessing

For this study, we picked an area in Southern California. The study area is displayed in Figure 1, and exists on the southwestern border of Escondido, California, precisely on the WUI where fire hazard is high. We chose this location because it contains a diversity of Southern Californian vegetation, varied topography, water bodies, and a spectrum of dense to sparse urban development. The Cocos Fire occurred in this study area in May 2014 as well, which provides the opportunity to test deep learning techniques on recently burned vegetation. The left-hand image shows the entire study extent. The right hand image of Figure 1 shows the region that was chosen for hand-labeling. The deep learning training region is in the the lower right box, and the test area as a combination of the northern and western image strips. These training and testing areas were defined to maintain as similar a distribution of land cover types between the train and test areas as possible.

As mentioned in Section II, existing vegetation products derived from satellite are highly variable, possibly due to the low spatial resolution of Landsat and the infrequent rate of publication of these products. LANDFIRE publishes a vegetation and fuels map for the continuous United States every two years. At the time of this writing, Planet imagery [20] is collected almost daily, at 3-5m/pixel. While it has fewer spectral bands, the spatial resolution increases the accuracy of what we identify. Additionally, as more frequent imagery is made available with satellites like those from Planet, this pipeline can be used to monitor and account for rapid land cover change due to environmental and social events. Our approach tests the use of the Planet satellite data on a Landsat-based workflow. As we are testing our results against Landsat-derived products, we aggregate the Planet data to the 30m resolution of these Landsat-derived products.

We collected imagery from Landsat (30m) and Planet's Planetscope (5m) sensors over the study area, each collected from the same month and each image cloud free. We preprocessed each scene by projecting each to the same map projection. We then chose a subset of the Escondido area to hand label each pixel. We clipped each scene to the same extent of this subset area, then tiled the imagery using commands from the

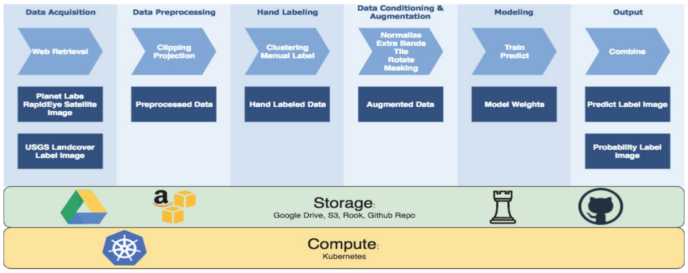

Fig. 2: End to end satellite data ingestion and analysis pipeline using deep learning.

Geospatial Data Abstraction Library (GDAL) [21]. Each tile generated had the same extent of 30m x 30m. Landsat tiles were one pixel, and the Planet tiles were 6x6 pixels.

We then collected the NLCD Land Cover map for the same area for 2011, the most recently published year. The map's resolution is 30m/pixel. These first two steps of our satellite image processing pipeline are illustrated in Figure 2 as *Data Acquisition* and *Data Preprocessing*.

B. Label Generation

One of the greatest challenges was using the land cover datasets derived from satellite as labels for machine learning as none of the existing products have great accuracy. The NLCD labels have inconsistencies, because it was generated from 2011 data and is a generalized product for the conterminous US; thus, we could not use this data for training labels. Instead, we chose to hand label the entire region, and use NLCD land cover categories as a guideline. This is indicated as the *Hand Labeling* step in Figure 2.

There were approximately one million pixels in the study area. To speed the process of labeling, we performed clustering of the Planet pixels into 20 clusters. We reviewed the cluster to find a predominant land cover type, and then hand labeled the pixels within that cluster that deviated from the predominant land cover type. We created the hand labels by converting the cluster image centers into a point Shapefile and then manually editing groups of points or individual points and labeling them in QGIS and ArcGIS. We used Worldview 3 (0.5m) imagery as a basemap to validate our hand labels. After labeling the 5m Planet data, we pooled the 6x6 pixels in each tile to choose a predominant land cover type for that 30m resolution tile. See Figure 3 for a comparison of the original NLCD labels, 5m hand labels, and pooled 30m hand labels for the area that we hand-labeled.

The discrepancy between our labels and the NLCD labels can be explained by the labeling approach. The NLCD map identifies a pixel classification by the dominant land cover type in that 30m pixel. Note the NLCD image on the right of Figure 3. The eastern half of the image has a lot of developed open space and developed low intensity areas, colored in pink. In our approach, we labeled each 5m pixel in Planet imagery, thus explicitly identifying the vegetation between buildings and neighborhoods, and aggregated the 5m labels to create 30m labels using a simple majority rule. We explicitly labeled regions that are vegetated as such, even if the vegetation is there because it was put there in a suburban context. Therefore, there are more details identifiable in our layers, and more explicit distribution of the vegetation versus the NLCD labels. To further show the need for refined labels, some of the NLCD labels in Figure 3 are incorrect. Note the reservoir in the southern center part of the scene. The center of the reservoir is classified as partly developed.

C. Deep Learning Models

We use CNNs and U-Nets, two types of deep learning models that have been successfully applied to image data.

A *CNN* is made up of different types of layers [22]–[24]. A CNN model used for classification typically consists of several blocks of convolutional layers followed by a pooling layer. Batch normalization and dropout layers can also be added. The last pooling layer then feeds into a fully connected layer where class scores are computed for the final classification.

A *U-Net* [17] is a type of convolutional auto-encoder with an *encoding path* that performs feature extraction with convolutional and pooling layers, followed by a *decoding path* that performs segmentation with upsampling and convolutional layers. Skip connections concatenate features from an encoding layer to the corresponding decoding layer, allowing for higher-resolution features to be combined with contextual information in generating output for the next layer in the decoding path.

Preparing the data for the models, and training and testing them are indicated as *Data Conditioning and Augmentation* and *Modeling* in Figure 2, and are detailed in the next section.

IV. Experimental Setup and Results

We built a CNN model and a U-Net model to predict the hand-labeled categories. The train area with 25,685 pixels was used to train the models, and both top and left test areas (with 13,547 pixels together) were used to test generalization.

A. CNN Model

For the CNN, we initially experimented with pre-trained CNNs and transfer learning. The use of CNNs trained on ImageNet data for transfer learning to other image data has proven to be very successful [25], [26]. We tested the use of transfer learning to our hand-labeled categories with the ResNet-50 [27] and VGG-19 [28] models. Results with the pre-trained models were poor, however, since the input images, at 6x6 pixels, had to be resized eight times for the VGG-19 and 36 times for the ResNet-50.

We then proceeded to build our own custom CNN. After some experimentation, we decided on the architecture in Table I.

We also tested different combinations of spectral bands in the input images. In addition to the five bands native to the Planet data (Section III-A), we also used these additional derived bands: EVI (Enhanced Vegetation Index), CCCI (Canopy Chlorophyll Content Index), and SAVI (Soil Adjusted Vegetation Index). These are vegetation indices commonly used in remote sensing applications with natural materials [29]. Our

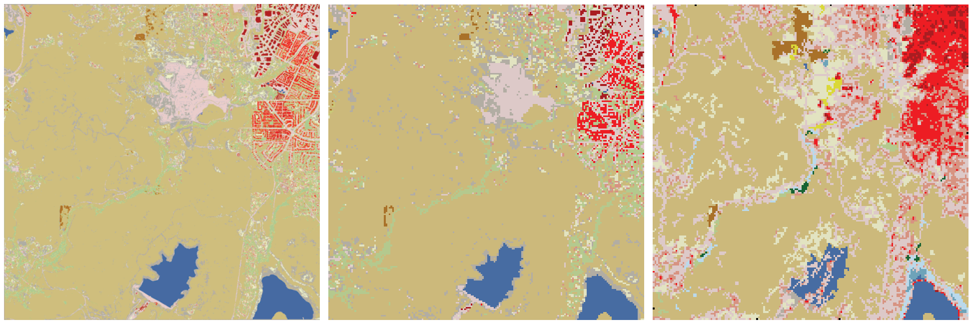

Fig. 3: Comparison of 5m hand labels, pooled 30m hand labels, and original NLCD labels for the hand-labeled area.

TABLE I: Custom CNN Architecture

Layer	Parameters
Convolutional	32-3x3 filters, ReLU
Convolutional	32-3x3 filters, ReLU
Pooling	32-2x2 filters, Max
Batch Normalization	
Convolutional	64-3x3 filters, ReLU
Pooling	64-2x2 filters, Max
Dropout	0.25 rate
Fully Connected	256 outputs, ReLU
Dropout	0.25 rate
Fully Connected	10 outputs, ReLU

tests indicated that using the five Planet bands along with EVI provided best results; thus, we used these six bands for all CNN experiments.

Image rotation was not used since only marginal prediction improvement was seen in preliminary experiments. The train area was split into train and validation datasets with a 80:20 ratio using stratified sampling. The CNN was trained with categorical cross entropy as the loss function, and a batch size of 128 using the SGD optimizer [30]. The best model was selected based on the lowest validation error.

B. U-Net Model

Preliminary testing indicated that multinomial classification with the U-Net yielded poor performance. We therefore decided to build a set of U-Net models instead, with a separate model trained on binary classification for each land cover category.

Each U-Net was trained to perform image segmentation for a specific land cover class; that is, the model was trained to output a binary mask with one for pixels belonging to that class, and zero otherwise. The general configuration of the various U-Net models was inspired by the winner of the Dstl Satellite Imagery Competition [19], and is described in Table II. The convolutional layers used the exponential linear unit (ELU) activation function, which is a modified version of ReLU that has been shown to speed up learning in deep networks [31].

Since each U-Net was trained separately and on a different class, we experimented with different numbers of layers and numbers of feature maps for each model. For a U-Net with three layers, the feature-map-per-layer configuration was 8-16-32, meaning that the first, second, and third layer had 8, 16, and 32 feature maps, respectively. We also tested four layers with

TABLE II: Baseline U-Net Architecture

Contracting Path	
Layer	Parameters
Convolutional	3x3 filters
Batch Normalization	
Dropout	0.1 rate
Convolutional	3x3 filters
Batch Normalization	
Pooling	2x2 filters, Max
Expanding Path	
Layer	Parameters
Transposed Convolutional	2x2 filters
Skip connections	
Convolutional	3x3 filters
Batch Normalization	
Droupout	0.1 rate
Convolutional	3x3 filters
Batch Normalization	

8-16-32-64 feature maps, and five layers with 8-16-32-64-128 feature maps.

Additionally, as with the CNN, we tested different combinations of spectral bands to use for each U-Net model. From these preliminary tests with model depth and spectral bands, the final U-Net model architecture was selected for each class as listed in Table III. Here, 'PL-5' indicates the original five Planet bands; 'RGB' indicates red, green, and blue bands; and 'Derived' indicates derived bands EVI, CCCI, and SAVI.

TABLE III: Architectures of U-Net Models

Category	# Layers	Spectral Bands
11-Water	3	PL-5
21-Dev Open Space	4	PL-5
22-Dev Low Intensity	3	RGB
23-Dev Medium Intensity	3	Derived
24-Dev High Intensity	3	PL-5
31-Barren Land	4	PL-5
43-Mixed Forest	4	PL-5
52-Shrub/Scrub	3	Derived
71-Grassland/Herbaceous	3	PL-5
82-Cultivated Crops	4	Derived

Preliminary analysis also indicated that image rotation helped with segmentation results, so data was augmented using rotation of 0, 90, 180, and 270 degrees. Image tiles of size 48x48 pixels with a padding of 16 were used as input to each U-Net. Since these are Planet images, the spatial resolution is 5m.

Each model was trained to output the binary mask of size

48x48 pixels for a specific land cover class. To combine predictions from all binary U-Net models for the final prediction, a simple max operation was performed on all U-Net predictions to determine the class for a pixel. In addition, to compare to the original Landsat land cover labels and also to the CNN output, the 5m predictions in the binary mask were pooled, using simple majority, to create a 30m binary mask.

The dice loss was used as the loss function to train the U-Net models. The dice loss is based on the dice similarity coefficient (DSC) [32], and is defined as follows:

$$\mathrm{DSC} = \frac{2 * |Y_{\text{true}} \bigcap Y_{\text{pred}}|}{|Y_{\text{true}}| + |Y_{\text{pred}}|}$$

$$\mathrm{DSC\ Loss} = 1 - \mathrm{DSC}$$

where Y_{true} are the true labels, Y_{pred} are the predictions for the land cover categories, and $\bigcap$ is the set intersection. The dice coefficient compares the similarity of two datasets, and is commonly used as a performance metric in image segmentation.

The train area was split into train and validation datasets with a 80:20 ratio. The models were trained with a batch size of 128 and the Nesterov Adam optimizer [30], and the best model was selected based on the lowest validation error.

C. System Setup

Our deep learning models were trained and tested using the Cognitive Hardware And Software Ecosystem Community Infrastructure (CHASE-CI), which is managed by the container orchestration system Kubernetes [33]. Our environment consisted of Ubuntu containers running interactive Jupyter notebooks. We used the Keras library [34] with the TensorFlow [35] backend to implement, train, and evaluate our models. We also used GDAL for processing the satellite imagery, and Spark [36] for clustering.

D. Experimental Results

Results of classifying the land cover categories for the CNN and U-Net are detailed in Figure 4 and Tables IV and V.

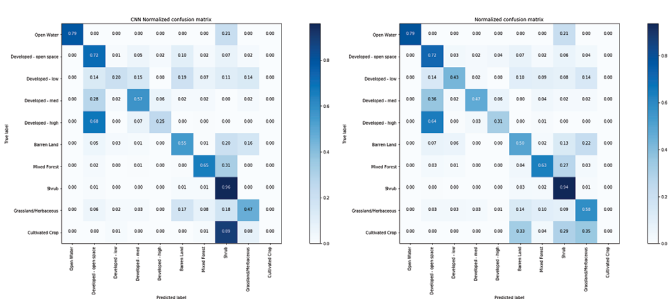

Fig. 4: Confusion matrices for CNN (left) and U-Net (right) on combined test areas

Breaking down results on individual land cover categories, we observed the following:

- Water was classified well by both models. This is not surprising since Water pixels have very different visible characteristics from the other categories.

TABLE IV: CNN Performance Metrics

Category	Precision	Recall	F1-Score	Support
11-Water	0.94	0.79	0.86	19
21-Dev Open Space	0.67	0.72	0.69	911
22-Dev Low Intensity	0.39	0.20	0.26	122
23-Dev Medium Intensity	0.19	0.57	0.29	47
24-Dev High Intensity	0.71	0.25	0.37	201
31-Barren Land	0.56	0.55	0.55	730
43-Mixed Forest	0.68	0.65	0.66	682
52-Shrub/Scrub	0.95	0.96	0.95	10,474
71-Grassland/Herbaceous	0.43	0.47	0.45	305
82-Cultivated Crops	0.00	0.00	0.00	83
Weighted Average/Total	0.87	0.87	0.87	13,574

TABLE V: U-Net Performance Metrics

Category	Precision	Recall	F1-Score	Support
11-Water	0.94	0.79	0.86	19
21-Dev Open Space	0.70	0.72	0.71	911
22-Dev Low Intensity	0.30	0.43	0.35	122
23-Dev Medium Intensity	0.34	0.47	0.39	47
24-Dev High Intensity	0.62	0.31	0.42	201
31-Barren Land	0.45	0.50	0.48	730
43-Mixed Forest	0.59	0.63	0.61	682
52-Shrub/Scrub	0.96	0.94	0.95	10,474
71-Grassland/Herbaceous	0.33	0.58	0.42	305
82-Cultivated Crops	0.00	0.00	0.00	83
Weighted Average/Total	0.86	0.85	0.86	13,574

- The F1-score was highest for Shrub in both models. This was the most abundant category, so the models saw plenty of examples of Shrub during training. Other vegetation types (e.g., Mixed Forest) were misclassified as Shrub, however. This is likely due to the similarity in appearance in these categories as well as the abundance of Shrub samples and their proximity to other vegetation types.
- The Developed-Low/Med/High categories had relatively low F1-scores and were often misclassified as Developed-Open Space. These categories look very similar and are physically close together, and can be easily confused even with the human eye. There were also very few samples of each category in the training set, less that 4% each.
- Neither model was able to correctly classify Cultivated Crops. This was due to the fact that there were only 7 samples of this category, out of the total 18,496 train samples. It is interesting to note that the CNN misclassified Cultivated Crops mostly as a single category, whereas the U-Net's misclassifications were spread out over several categories. This may be because the CNN was trained on multiclass classification with softmax, forcing it to choose a single category, while the U-Net models were trained on binary classification, which can allow several models to produce similar classification scores on difficult classes.

In general, performance results of the CNN and U-Net were very similar, as seen by the F1-scores in Figure 5. The CNN performed better than the U-Net on some categories, notably Barren Land, Mixed Forest, and Grassland, while the U-Net performed better on the Developed categories. Overall, however, there was no significant difference in performance, as can also be seen in Figure 4 and Tables IV and V.

Figure 6 shows how the CNN and U-Net predictions compare

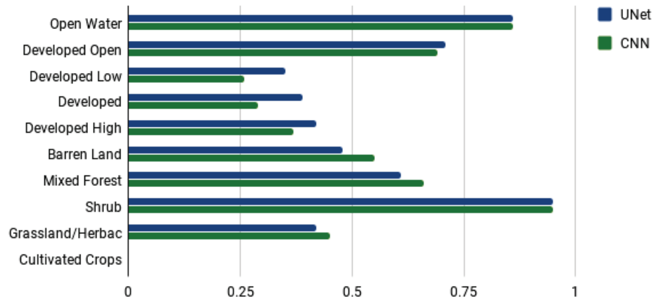

Fig. 5: CNN vs. U-Net F1-scores on combined test areas

to the true hand labels for the top test area. The brown pixels seen in the Hand Labels image are Cultivated Crops, which is the category that both models were not able to identify due to the small number of training samples for this class. Thus, there are Cultivated Crops pixels in the NLCD image and none in the CNN or U-Net results. The extra red pixels seen in the CNN Predictions image as compared to the Hand Labels image in the middle right of the image mark the misclassifications by the CNN of Developed Low Intensity to Developed Medium Intensity, which is also indicated in the CNN confusion matrix in Figure 4. The light yellow and gray pixels to the right of the brown pixels represent Grassland and Barren Land, respectively. It can be seen that the CNN's predictions more closely match the hand labels here, which also agrees with results in Figure 5.

(a) 30m Hand Labels

(b) NLCD Labels

(c) CNN Predictions

(d) U-Net Predictions

Fig. 6: Comparison of 30m hand labels, NLCD labels, CNN predictions, and U-Net predictions for top test area

We applied the CNN and U-Net models to the entire study area. That is, we used each model to predict land cover categories for the entire original area described in Section III. The results are shown in Figure 7 along with the original NLCD labels for this area. As can be seen, category predictions for the CNN and U-Net are very similar. Further, it is important to note that these predictions refine the NLCD labels to a more realistic picture of what is actually on the ground. This observation validates the predictive accuracy of our models and indicates that they can be used to generate target labels for other, larger areas.

V. Conclusions and Future Work

This paper presents an approach using deep learning to generate land cover maps from satellite imagery. Our approach is unique in that it applies a pixel-based approach to high resolution imagery to classify land cover. Our goal is to process land classifications that better depict vegetation types in the WUI, where land cover changes rapidly, and where the location of interspersed vegetation and homes are critical for understanding fire behavior. The results presented in this paper indicate that our classifications would do much better at representing the distribution of vegetation than the NLCD model in this part of Escondido, California.

As a part of future work to improve the prediction performance of the models we plan to investigate the use of more data augmentation techniques such as shift and zoom. This will add more variability to the training data and enable the models to be more robust to slight changes in the input image. Additionally, many of the smaller classes yielded worse results than larger classes, suggesting that addressing class imbalance may lead to improved classification performance. Incorporating other spectral bands, e.g., infrared and short-wave infrared, will provide additional input information to the models, and should also help to improve performance. These bands, with longer wavelengths than visible light, can help to separate vegetation from non-vegetation materials as well as different types of urban materials such as concrete and tile to help further distinguish between the Developed categories.

Our models can now be used to automate the process of labeling pixels in other areas with similar land cover. To improve the accuracy of the current models, we can examine predictions that have low scores or small differences between the top N scores, which indicate predictions with low confidence, and re-label them as necessary. This can be considered a type of active learning to selectively choose samples to guide training. The new labels can then be used to refine existing models to generate more accurate predictions, which can be used to fully automate the labeling process.

Since hand labeling is a time-consuming, tedious, and error-prone process, the ability to use our models to automate the labeling process adds scalability and accuracy to our analytics pipeline. This is essential as we proceed with applying our approach to larger and more diverse areas.

Acknowledgements

This work was supported in part by NSF-1331615 under CI, Information Technology Research, and SEES Hazards programs and NSF-1730158 for Cognitive Hardware and Software Ecosystem Community Infrastructure (CHASE-CI).

The authors also appreciate the collaboration and in-kind support of Jenny Palomino, Planet, Digital Globe, the Google Earth Engine, and Pradeep Saddi.

References

[1] Ilkay Altintas, Jessica Block, Raymond de Callafon, Daniel Crawl, Charles Cowart, Amarnath Gupta, Mai Nguyen, Hans-Werner Braun, Jürgen P. Schulze, Michael Gollner, Arnaud Trouve, and Larry Smarr, "Towards an Integrated Cyberinfrastructure for Scalable Data-driven

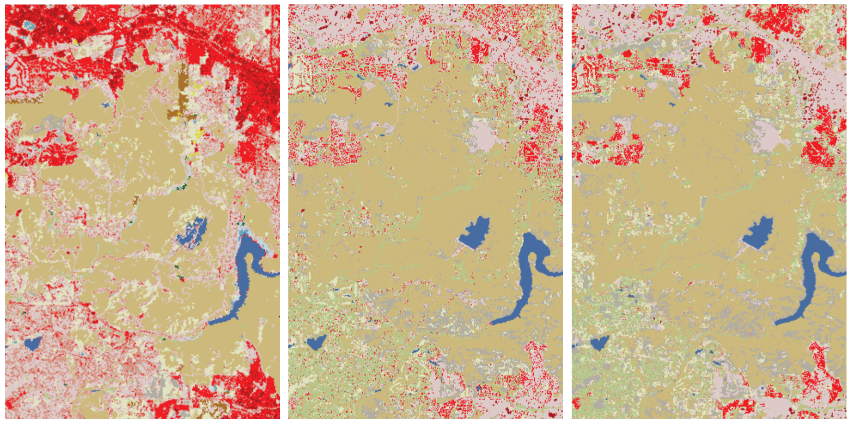

Fig. 7: Comparing NLCD labels, CNN predictions, and U-Net predictions for entire study area.

Monitoring, Dynamic Prediction and Resilience of Wildfires," in *Proc. of the Int. Conf. on Computational Science, ICCS 2015*, 2015, pp. 1633–1642.

[2] "The WIFIRE project website.," https://wifire.ucsd.edu, 2018.

[3] J. Block, M. Yazdani, M. Nguyen, D. Crawl, M. Jankowska, J. Graham, T. DeFanti, and I. Altintas, "An unsupervised deep learning approach for satellite image analysis with applications in demographic analysis," in *2017 IEEE 13th International Conference on e-Science (e-Science)*, Oct 2017, pp. 9–18.

[4] "NASA Landsat Website," https://landsat.gsfc.nasa.gov, 2017.

[5] "LANDFIRE Website," https://www.landfire.gov, 2018.

[6] "The Forest Inventory Analysis Website," https://daac.ornl.gov/NACP/guides/NAFD-NEX_Forest_Disturbance.html, 2018.

[7] C.G. Homer, J.A. Dewitz, L. Yang, S. Jin, P. Danielson, G. Xian, J. Coulston, N.D. Herold, J.D. Wickham, and K. Megown, "Completion of the 2011 national land cover database for the conterminous united states-representing a decade of land cover change information," *Photogrammetric Engineering and Remote Sensing*, vol. 81, no. 5, pp. 345–354, 2015.

[8] "The North American Forest Dynamics Website," https://www.fia.fs.fed.us/forestcarbon/index.php, 2018.

[9] M. C. Hansen, P. V. Potapov, R. Moore, M. Hancher, S. A. Turubanova, A. Tyukavina, D. Thau, S. V. Stehman, S. J. Goetz, T. R. Loveland, A. Kommareddy, A. Egorov, L. Chini, C. O. Justice, and J. R. G. Townshend, "High-resolution global maps of 21st-century forest cover change," *Science*, vol. 342, no. 6160, pp. 850–853, 2013.

[10] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al., "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[13] Nataliia Kussul, Mykola Lavreniuk, Sergii Skakun, and Andrii Shelestov, "Deep learning classification of land cover and crop types using remote sensing data," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 5, pp. 778–782, 2017.

[14] Yao Yao, Haolin Liang, Xia Li, Jinbao Zhang, and Jialv He, "Sensing urban land-use patterns by integrating google tensorflow and scene-classification models," *arXiv preprint arXiv:1708.01580*, 2017.

[15] Guang Xu, Xuan Zhu, Dongjie Fu, Jinwei Dong, and Xiangming Xiao, "Automatic land cover classification of geo-tagged field photos by deep learning," *Environmental Modelling & Software*, vol. 91, pp. 127–134, 2017.

[16] Adrian Albert, Jasleen Kaur, and Marta C Gonzalez, "Using convolutional networks and satellite imagery to identify patterns in urban environments at a large scale," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 1357–1366.

[17] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[18] Alberto Garcia-Garcia, Sergio Orts-Escolano, Sergiu Oprea, Victor Villena-Martinez, and Jose Garcia-Rodriguez, "A review on deep learning techniques applied to semantic segmentation," *arXiv preprint arXiv:1704.06857*, 2017.

[19] "Dstl satellite imagery competition, 1st place winner's interview: Kyle lee," http://blog.kaggle.com/2017/04/26/dstl-satellite-imagery-competition-1st-place-winners-interview-kyle-lee/, 2017.

[20] "Planet Website," https://www.planet.com/, 2018.

[21] "The gdal website," GeospatialDataAbstractionLibrary, 2018.

[22] Andrej Karpathy, "CS231n convolutional neural networks for visual recognition ," http://cs231n.github.io/convolutional-networks/.

[23] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[24] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[25] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson, "CNN features off-the-shelf: an astounding baseline for recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2014, pp. 806–813.

[26] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson, "How transferable are features in deep neural networks?," in *Advances in neural information processing systems*, 2014, pp. 3320–3328.

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[28] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[29] "Index database: A database for remote sensing indices," https://www.indexdatabase.de/db/s-single.php?id=10, 2017.

[30] "Keras documentation: Usage of optimizers," https://keras.io/optimizers/#usage-of-optimizers.

[31] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, 2015.

[32] Lee R Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.

[33] "Kubernetes: Production-Grade Container Orchestration," https://kubernetes.io/, 2018.

[34] "Keras: The Python Deep Learning Library ," https://keras.io/.

[35] "TensorFlow," https://www.tensorflow.org/.

[36] "Apache Spark - Unified Analytics Engine for Big Data," http://spark.apache.org, 2018.