Cloud Based Automatic Building and Road Extraction from Large Scale Open Geospatial Datasets

Yunzhi Shi, Tianyu Zhang, Daniel Hogan, Jake Shermeyer, Adam Van Etten, Xin Chen

October 19, 2020

1 Introduction

We propose a half day tutorial at WACV 2021 focused on infrastructure identification from open geospatial datasets. This proposal is a collaboration between the AWS Machine Learning Solutions Lab and CosmiQ Works teams, with CosmiQ focusing on the datasets, algorithms, and applications, while the AWS Machine Learning Solutions Lab team will focus on cloud implementation and scaling of algorithms. Details about the proposal team and course implementation are as follows.

2 Proposal Team

- Yunzhi Shi, Data Scientist, AWS ML Solutions Lab, `shiyunzh@amazon.com`. Yunzhi helps AWS customers address business problems with AI and cloud capabilities. Recently, he has been building CV, search, and forecast solutions for various customers. Prior to Amazon, Yunzhi obtained his Ph.D. in Geophysics from The University of Texa at Austin.

- Tianyu Zhang, Data Scientist, AWS ML Solutions Lab, `ttizha@amazon.com`. Tianyu helps AWS customers solve business problems by applying ML and AI techniques. Most recently, he has built NLP model and predictive model for procurement and sports.

- Daniel Hogan, Data Scientist, In-Q-Tel CosmiQ Works `dhogan@iqt.org`. Daniel is a data scientist with a geospatial focus. His research has looked at dataset development and synthetic aperture radar. Daniel received a Ph.D. in Physics from the University of California, Berkeley.

- Jake Shermeyer, Research Scientist, In-Q-Tel CosmiQ Works `jshermeyer@iqt.org`. Jake is a researcher and geographer specializing in geospatial machine learning and computer vision. His research with satellite imagery focuses on time series analysis, super-resolution, the value of synthetic data, and object detection. Jake served as the lead for SpaceNet 6, a sensor fusion challenge and dataset focused on both synthetic aperture radar and electro-optical remote sensing data and their application to foundational mapping problems.

- Adam Van Etten, Chief Data Scientist, In-Q-Tel, avanetten@iqt.org. Adam focuses on applied machine learning topics of interest to the US Government. His most recent research lies in the geospatial analytics realm, where he applies machine learning and computer vision techniques to satellite imaging data. Other recent foci for Adam are helping run the SpaceNet initiative, and exploring the limitations and utility functions of machine learning techniques.

- Xin Chen, Senior Manager, AWS ML Solutions Lab, xcaa@amazon.com. Xin leads his team to help AWS customers identify and build machine learning solutions to address their organization's high-est return-on-investment machine learning opportunities. Prior to Amazon, Xin was a Director of Engineering at HERE Technologies whose team completed pioneering work to achieve the automation of next generation map creation using computer vision and machine learning technologies. Xin is an adjunct faculty at Northwestern U. and Illinois Institute of Technology.

3 Course Description

The course will consist of five sections (plus a break), organized as follows.

1. **SpaceNet Dataset, Algorithms, Applications (80 minutes)** In the first section, we will introduce the SpaceNet [ELB18] dataset, along with open source algorithms developed from this dataset and discuss applications. The SpaceNet dataset is a large corpus of imagery and labels that is hosted as an Amazon Web Services (AWS) Public Dataset. It contains 70,000 square km of high-resolution imagery, 11,000,000 building footprints, and 20,000 km of road labels to ensure that there is adequate open source data available for geospatial machine learning research. Seven public data science challenges have been run with this data, tackling various problems from building footprint extraction to road travel time prediction to urban change detection. The winning algorithms of these challenges are open source, and address a whole host of humanitarian use cases (disaster response, evacuation planning, urban planning, etc) that we will discuss in detail.

2. **Synthetic Data and Rare Objects (35 minutes)** The second section will focus on the Rareplanes dataset and study. RarePlanes is a unique open-source machine learning dataset that incorporates both real and synthetically generated satellite imagery, and is the largest openly-available

high resolution dataset built to test the value of synthetic data from an overhead perspective. The real portion of the dataset consists of $>$ 250 satellite images spanning $>$ 100 locations with 15,000 hand-annotated aircraft. The accompanying synthetic dataset features 50,000 synthetic satellite images with 600,000 aircraft annotations. Both the real and synthetically generated aircraft feature fine grain attributes such as length, wingspan, engine type, etc. We conduct extensive experiments to evaluate the real and synthetic datasets and compare performances, and show the value of synthetic data for the task of detecting and classifying aircraft from an overhead perspective. The lessons learned from this study translate readily to other objects and modalities.

3. **5 minute break**

4. **Cloud Services (25 minutes)** In this section we will talk about Amazon SageMaker, a fully managed ML service that provides every developer and data scientist with the ability to build, train, and deploy ML models quickly. Amazon SageMaker Ground Truth is a data labeling service that makes it easy to build highly accurate training dataset in the data preparing step. Amazon SageMaker Notebook Instance is the ML compute instance running the Jupyter Notebook APP, offering a ML development environment that allows users to prepare and process data, write code to train, deploy and validate models. The SageMaker also provides several images of built-in ML algorithms that makes the training process much smoother and simpler. In the training job, Amazon SageMaker Hyperparameter Tuning helps to tune the hyperparameters and find the best version of a model automatically. After training, Amazon SageMaker can deploy the trained model into production with a single click so that it can start generating predictions for real-time or batch data and monitor the performance of model. (Tianyu)

5. **Cloud Notebooks (75 minutes)** In this section, we will walk through deep learning models that extract building footprints and road networks using Jupyter notebooks developed by AWS Machine Learning Solutions Lab team. The notebooks reproduce winning algorithms from the SpaceNet challenges. In addition to the SpaceNet satellite images [ELB18], we introduce USGS 3D Elevation Program (3DEP) light detection and rangin (LiDAR) data to the workflow. We demonstrate using satellite images, LiDAR data, or combination of both to train and test deep learning models for map feature extractions. This tutorial shares the notebooks and provides instructions on running ML services on large scale geospatial data on Amazon SageMaker. At the end of this section, audiences can reproduce the notebook content, apply the models to other area of interests, and innovate with new ideas to improve. The audiences can also appreciate the benefits of cloud computing and storage first-hand. (Yunzhi)

6. **Summary and Conclusions (10 minutes)**

4 Related Works

The SpaceNet dataset and challenge was featured in CVPR EarthVision 2017, 2019, and 2020. The authors of this proposal also helped organized the DeepGlobe Workshop at CVPR 2018, which used SpaceNet data. Our proposed tutorial directly follows upon these previous workshops, with the added layer of focusing on the applications of computer vision by deploying models in cloud environments.

References

[ELB18] Adam Van Etten, Dave Lindenbaum, and Todd M. Bacastow. "SpaceNet: A Remote Sensing Dataset and Challenge Series". In: *CoRR* abs/1807.01232 (2018). arXiv: `1807.01232`. URL: `http://arxiv.org/abs/1807.01232`.

5 Appendix

Attachment 1: Technical abstract by AWS Machine Learning Solutions Lab, "Cloud Based Automatic Building and Road Extraction from Large Scale Open Geospatial Datasets"

Cloud Based Automatic Building and Road Extraction from Open Datasets of Satellite and LiDAR

Yunzhi Shi
Amazon Web Services
Austin, TX USA
shiyunzh@amazon.com

Xin Chen
Amazon Web Services
Chicago, IL USA
xcaa@amazon.com

Tianyu Zhang
Amazon Web Services
Austin, TX USA
ttizha@amazon.com

ABSTRACT

We author Jupyter notebooks to develop deep learning models on Amazon SageMaker instance. These models automatically extract building footprints and road networks from open geospatial datasets. The notebooks reproduce winning algorithms from the SpaceNet challenges. In addition to the SpaceNet satellite images, we introduce USGS 3D Elevation Program (3DEP) light detection and ranging (LiDAR) data to the workflow. We demonstrate using satellite images, LiDAR data, or combination of both to train and test deep learning models for building and road extraction. Both datasets are hosted on Amazon Web Services (AWS).

This tutorial will share the notebooks and provide hands-on and step-by-step instructions on running machine learning services to extract features from large scale geospatial data on AWS. At the end of the tutorial, audiences can reproduce the building and road extraction tasks, apply the models to other area of interests where satellite or LiDAR data are available, and innovate with new ideas to improve the performances. The audiences can also appreciate the benefits of cloud computing and storage first-hand.

KEYWORDS

Satellite photo, LiDAR, AWS, SageMaker, buildings, road network, USGS, SpaceNet, Deep Learning

1 INTRODUCTION

Sharing data in the cloud lets data users spend more time on data analysis rather than data acquisition. The Registry of Open Data on AWS [1] is a service that helps people discover and share datasets that are available via AWS resources. When data is shared on AWS, anyone can analyze it and build services on top of it using a broad range of computing and analytics products, including Amazon EC2, Amazon Athena, AWS Lambda, and Amazon EMR. It develops new cloud-native techniques, formats, and tools. It also encourages the development of communities that benefit from access to shared datasets.

As for geospatial domain, AWS Open Data Registry includes several datasets suitable for machine learning research. For example, SpaceNet [2] was launched in August 2016 as an open innovation project offering a repository of freely available imagery with co-registered map features. The SpaceNet partners also launched a series of public prize competitions to encourage improvement of remote sensing machine learning algorithms. Another dataset is USGS 3DEP LiDAR data [3]. Its goal is to complete acquisition of nationwide LiDAR to provide the first ever national baseline of consistent high-resolution topographic elevation data, collected in a timeframe less than a decade.

Today, map features such as building footprints, road networks, and points of interest are primarily created through manual techniques. Advancing automated feature extraction techniques will serve important downstream users of map data including humanitarian and disaster response [2]. Furthermore, we believe solving this challenge is an important steppingstone to unleash the power of advanced computer vision algorithms applied to a variety of remote sensing data applications.

We author Jupyter notebooks of automatic building and road extraction using deep learning techniques. We reproduce winning algorithms from SpaceNet challenges, and combine both SpaceNet satellite image and USGS LiDAR data to train and evaluate model performances. We demonstrate the model accuracy improvement by introducing LiDAR data. The tutorial will share the notebooks with the audiences and provide hands-on instructions.

The target audiences are both academics and industry data scientists who are interested in learning to use machine learning services on AWS, getting hands-on experience of running large scale feature extraction from geospatial datasets. Audience is recommended to create an AWS account to follow along; “lite” version notebooks are available to run with free-tier services.

2 DATASETS

2.1 SpaceNet Dataset

SpaceNet data is a large corpus of labeled satellite imagery published by the project partners and hosted on AWS. The project also launched a series of public prize competitions ranging from automatic building extraction [4–6], road extraction [7,8], and recently published multi-temporal urban development analysis [9]. The dataset covers 11 area of interests (AOIs), including Rio de Janeiro, Las Vegas, Paris, etc. Take Las Vegas as example, the images in this AOI cover 216km^2 area, include 151367 building polygon labels and 3685km road labels.

2.2 USGS 3DEP LiDAR Dataset

The USGS 3DEP LiDAR dataset provides two realizations of the point cloud data. The first resource is a public repository in

Entwine Point Tiles format, which is a lossless, full density, streamable octree based on LASzip encoding. This format is suitable for online visualization [10]; Fig. 1 shows a visualization example in Las Vegas. The second resource is the in LAZ (compressed LAS) format with requester-pays access.

Figure 1: Visualization of USGS 3DEP LiDAR data in Las Vegas, hosted via Entwine Point Tiles format [10].

2.3 Data Registration

For this tutorial, we select the Las Vegas AOI where both SpaceNet satellite images and USGS LiDAR data are available. Among SpaceNet data categories, we use the 30cm resolution pan-sharpened 3-band RGB geotiff and corresponding building and road labels. To improve the visual feature extraction performance, we process the data by white balancing and convert to 8-bit (0–255) values for the ease of postprocessing. Fig. 2 shows the RGB value aggregated histogram of all images after the processing.

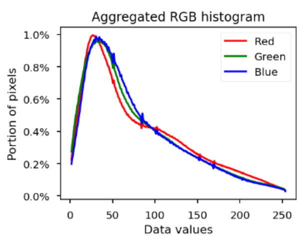

Figure 2: RGB value aggregated histogram of all images after the white balancing and 8-bit conversion.

While satellite images are 2D images, the USGS LiDAR data is 3D point cloud format and thus requires conversion and projection. We use Matlab and LAStools [11] to map each 3D LiDAR point to pixel-wise location corresponding to SpaceNet tiles, and generate two sets of attribute images: elevation and reflectivity intensity. The elevation ranges from ~2000–3000 feet, and the intensity ranges from 0–5000 units. Fig. 3 shows the aggregated histogram of all images for elevation and reflectivity intensity values.

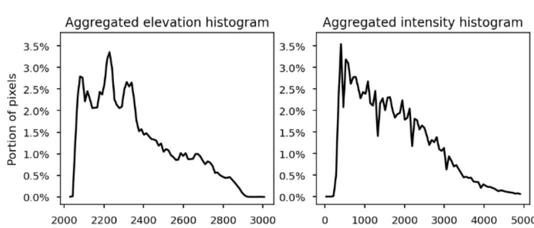

Figure 3: Aggregated histogram of all images for LiDAR elevation and reflectivity intensity values.

Finally, we merge either one of the LiDAR attributes and merge them with the RGB images. The images are saved in 16-bit since LiDAR attribute values can be larger than 255, the 8-bit upper limit. We make these processed and merged data available via AWS S3 bucket for this tutorial. Fig. 4 shows three sample images.

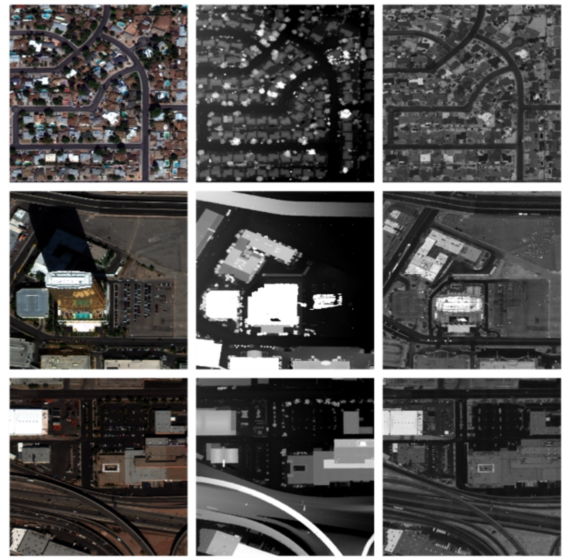

Figure 4: Three samples of merged RGB + LiDAR images. Columns from left to right: RGB image, LiDAR elevation attribute, and LiDAR reflectivity intensity attribute.

3 BUILDING EXTRACTION

The 1st and 2nd SpaceNet challenge [4,5] aimed to extract building footprints from the satellite images in various AOIs. The 4th SpaceNet challenge [6] posed similar task with more challenging off-nadir (i.e. oblique look angle) imagery. In this section, we reproduce a winning algorithm and evaluate its performance with both RGB images and LiDAR data.

3.1 Training Data

In the Las Vegas AOI, SpaceNet data is tiled to size 200m×200m. We locate 3084 tiles where both SpaceNet imagery and LiDAR data are available, and merge them together. Unfortunately, the labels of test data for scoring in the SpaceNet challenges are not published, so we split the merged data by 70%/30% for training and evaluation. We select elevation in this case because it is more representative to extract buildings than reflectivity intensity.

3.2 Model

We reproduce a winning algorithm from SpaceNet challenge 4 [6] by XD_XD. The model has a U-net [12] architecture with skip-connections between encoder and decoder, and a modified VGG16 [13] as backbone encoder. The model takes three different types of input: (1) 3-channel RGB image, same as the original contest, (2) 1-channel LiDAR elevation image, and (3) 4-channel RGB + LiDAR merged image. We will train three models and compare their performances in evaluation section.

The label for training is binary mask converted from polygon geojson by Solaris [14], a machine learning pipeline library developed by CosmiQ Works. We select a combined loss of binary cross-entropy and Jaccard loss with a weight factor $\alpha = 0.8$:

$$\mathcal{L} = \alpha\mathcal{L}_{\text{BCE}} + (1 - \alpha\mathcal{L}_{\text{Jaccard}})$$

The model is implemented with Solaris and deployed on an Amazon SageMaker p3.8xlarge instance (4× V100 GPUs). We train the models with batch size 20, Adam optimizer, and 10^{-4} learning rate for 100 epochs. Fig. 5 shows some examples of input image (RGB + LiDAR), predicted building mask output by training with both RGB and LiDAR data, and ground truth building mask.

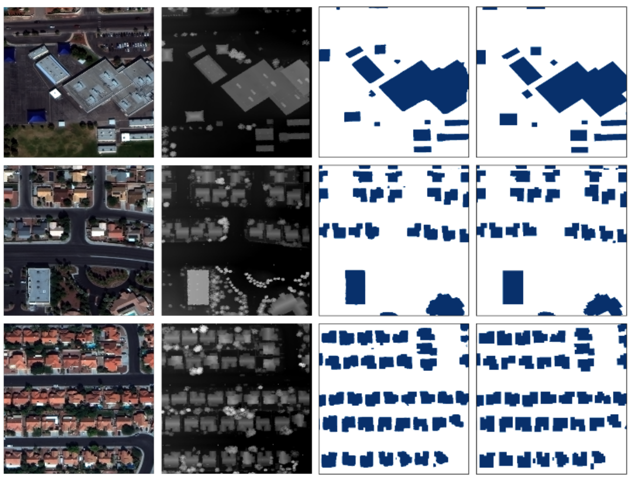

Figure 5: Examples of building extraction model inputs and outputs. Columns from left to right: RGB image, LiDAR elevation image, model prediction trained by both RGB and LiDAR data, and ground truth building footprint mask.

3.3 Evaluation

After model inference on the test dataset (30% hold-out), we evaluate the model performance using the same metric as in the original contest: aggregated F-1 score with intersection of union (IoU) ≥ 0.5 criterion. Table 1 shows the F-1 scores from three models trained with (1) RGB images, (2) LiDAR elevation images, and (3) RGB + LiDAR merged images. Compared to using RGB only as in the original SpaceNet competition, the model trained using only LiDAR elevation images can achieve score only a few percent worse. When combining both RGB and LiDAR elevation in training, the model outperforms RGB-only model. For reference, F-1 scores of top-3 teams from SpaceNet challenge 2 in this AOI are 0.885, 0.829, and 0.787 (we do not compare directly because they use a different test set for scoring).

Table 1: F-1 scores of building extraction models

Training data type	Aggregated F-1 scores
RGB images	0.82680
LiDAR elevation	0.80676
RGB + LiDAR merged	**0.85312**

4 ROAD EXTRACTION

The 3rd SpaceNet challenge [7] aimed to extract road networks from the satellite images, and the 5th SpaceNet challenge [8] add to the task to predict road speed along with the network extraction in order to minimize travel time and plan optimal routing. Similar to the previous section, we will reproduce a top algorithm, train different models with either RGB images, LiDAR attributes, or both of them, and evaluate their performances.

4.1 Training Data

The road network extraction uses larger tiles with size 400m×400m. We generate 918 merged tiles, and split by 70%/30% for training and evaluation. In this case, we select reflectivity intensity for road extraction because road surfaces often consist of materials that have distinctive reflectivity among background, e.g. paved surface, dirt road, asphalt.

4.2 Model

We reproduce the CRESI algorithm [15] for road networks extraction. It also has a U-net architecture but uses ResNet [16] as backbone encoder. Again, we train the model with three different types of input: (1) 3-channel RGB image, (2) 1-channel LiDAR intensity image, and (3) 4-channel RGB + LiDAR merged image.

To extract road location and speed together, binary road mask will not provide enough information for training. As mentioned in CRESI paper [15], we can convert speed metadata to either continuous mask (0–1 values) or multi-class binary mask. Because their test results show that multi-class binary mask perform better, we will use the latter conversion scheme. Fig. 6 and 7 show visualizations of the multi-class road mask.

We train the model with the same setup as in building extraction. Fig. 8 shows some examples of input image (RGB + LiDAR), predicted road mask output by training with both RGB and LiDAR data, and ground truth road mask.

Figure 6: Visualization of multi-class road mask. Left: RGB image tile. Right: road mask with color coding in which yellow-to-red colormap represents speed values from low to high speed (0–65 mph).

4.3 Evaluation

We implement the average path length similarity (APLS) score [17] to evaluate the road extraction performance. This metric is used in SpaceNet road challenges because APLS consider both logical topology (connections within road network) and physical topology (location of the road edges and nodes). The APLS can be weighted by either length or travel time, higher score means better performance.

Figure 7: Break down of the 8-class road masks. The first 7 binary masks represent road corresponds to 7 bins of speed within 0–65 mph. The 8th mask (bottom right) represent the aggregation of all previous masks.

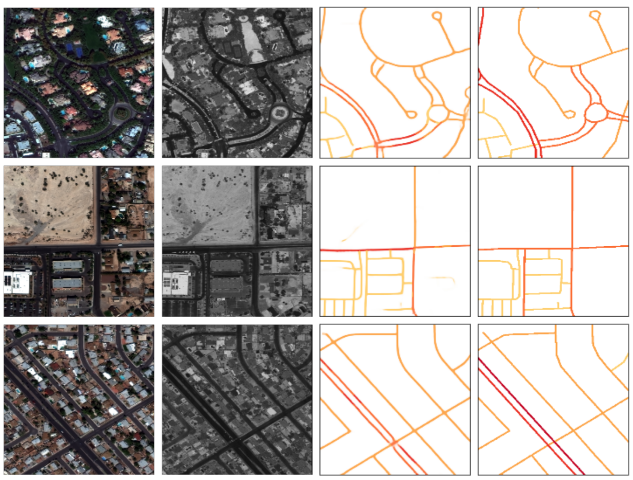

Figure 8: Examples of road extraction model inputs and outputs. Columns from left to right: RGB image, LiDAR reflectivity intensity image, model prediction trained by both RGB and LiDAR data, and ground truth road mask.

We convert multi-class road mask predictions to skeleton and speed-weighted graph and compute APLS scores. Table 2 shows the APLS scores of the three models. Similar to building extraction results, LiDAR-only result achieve close scores to RGB-only result, while RGB + LiDAR gives the best performance.

Table 2: APLS scores of road extraction models

Training data type	$APLS_{length}$	$APLS_{time}$
RGB images	0.59624	0.54298
LiDAR intensity	0.57811	0.52697
RGB + LiDAR merged	**0.63651**	**0.58518**

5 SUMMARY

We present reproductions of SpaceNet winning algorithms, implement machine learning models on Amazon SageMaker instances to automatically extract building and road from geospatial data. In addition to RGB satellite imagery, we process USGS 3DEP LiDAR data and incorporate the LiDAR attributes in those models. Using dataset in the Las Vegas AOI, we show LiDAR data can be used to perform the same task with similar accuracy, and outperform RGB models when combined with RGB imagery.

We prepare Jupyter notebooks and will share them in the tutorial to provide step-by-step guide. At the end of the tutorial, audiences can reproduce the building and road extraction tasks, apply the models to any other area of interests, and innovate with new ideas to improve the performances. The audiences can also appreciate the benefits of cloud computing and storage first-hand.

This tutorial teaches cloud computing in a large geospatial data analysis context, highlighting multimodal models that process both satellite image and LiDAR data. Our future work is to generate and share tooling on AWS to streamline the process of geospatial data.

ACKNOWLEDGMENTS

LiDAR data courtesy of U.S. Geological Survey.

REFERENCES

[1] "Registry of Open Data on AWS," [Online]. Available: https://registry.opendata.aws/.

[2] A. Van Etten, D. Lindenbaum and T. M. Bacastow, "Spacenet: A remote sensing dataset and challenge series," *arXiv preprint arXiv:1807.01232,* 2018.

[3] "USGS 3D Elevation Program," [Online]. Available: https://www.usgs.gov/core-science-systems/ngp/3dep.

[4] "SpaceNet Round 1 Challenge Implementations," 2017. [Online]. Available: https://github.com/SpaceNetChallenge/BuildingDetectors/.

[5] "SpaceNet Round 2 Challenge Implementations," 2017. [Online]. Available: https://github.com/SpaceNetChallenge/BuildingDetectors_Round2.

[6] "SpaceNet Round 4 Challenge Implementations," 2018. [Online]. Available: https://github.com/SpaceNetChallenge/SpaceNet_Off_Nadir_Solutions.

[7] "SpaceNet Round 3 Challenge Implementations," 2018. [Online]. Available: https://github.com/SpaceNetChallenge/RoadDetector.

[8] "SpaceNet Round 5 Challenge Implementations," 2019. [Online]. Available: https://github.com/SpaceNetChallenge/SpaceNet_Optimized_Routing_Solutions.

[9] "SpaceNet Round 7 Challenge," 2020. [Online]. Available: https://www.cosmiqworks.org/current-projects/spacenet-7/.

[10] "USGS & Entwine," [Online]. Available: https://usgs.entwine.io/.

[11] M. Isenburg, "LAStools-efficient LiDAR processing software," 2014.

[12] O. Ronneberger, P. Fischer and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, 2015.

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

[14] CosmiQ Works, "Solaris: An open source ML pipeline for overhead imagery," 2019. [Online]. Available: https://github.com/CosmiQ/solaris.

[15] A. Van Etten, "City-scale road extraction from satellite imagery v2: Road speeds and travel times," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020.

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

[17] CosmiQ Works, "CosmiQ/apls: Python code to evaluate the APLS metric," 2017. [Online]. Available: https://github.com/CosmiQ/apls.