Creating Geo-specific Synthetic Environments using Deep learning and Process Automation

Bodhiswatta Chatterjee, Hermann Brassard, Bhakti Patel
Presagis Inc. Canada
Montreal, Canada
Bodhiswatta.Chatterjee@presagis.com, Hermann.Brassard@presagis.com, Bhakti.Patel@presagis.com

ABSTRACT

Creation of geo-specific 3D environments for training and simulation requires a lot of information along with Electro-Optical (EO) imagery. Acquiring vectors of different object classes along with attributes for each vector is a labor-intensive process. Another important component for making the 3D environment geo-specific is the depth information obtained from Digital Surface Models (DSM), information which is often expensive, difficult to acquire, and might be noisy. This paper discusses how Deep Learning (DL) based techniques can be used for the extraction of attributed vectors of different object classes from EO imagery and eventually create geo-specific 3D synthetic environments without DSM data.

The contribution of this work is twofold: first, multi-level Deep Learning techniques are used for the extraction of building footprints and attributes (e.g., roof type) for each extracted building. Using extracted and derived features (area, shape, etc.), the building heights are estimated which alleviates the requirement of acquiring expensive and difficult to procure DSM data. Second, the challenge of creating the huge training datasets required to train Deep Learning models is addressed by generating synthetic data using our in-house software to solve the problem of roof type classification where no labeled training dataset exists. A performance-based comparative analysis of classification techniques on synthetic data with other state-of-the-art techniques like few-shot classification is done to provide insights on how synthetic/hybrid datasets can be used when labeled training datasets are not available.

Finally, a qualitative comparison of 3D reconstructions is performed where the models are created using our automated 3D reconstruction pipeline. The reconstructions will allow the comparison of the results obtained from human produced footprint with those produced through Artificial Intelligence (AI). The result will show reconstructions based on AI-inferred attributes is very close to geo-specific standards and offers the avenue to remove the labor-intensive manual attribution or acquisition of expensive DSM data.

ABOUT THE AUTHORS

Bodhiswatta Chatterjee is an Applied Researcher at Presagis Inc. Canada, Modeling and Simulation Division. He holds a Master's Degree in Computer Science from Concordia University, Canada focusing on Computer Vision and Machine Learning (Deep Neural Networks). At Presagis, Bodhiswatta uses his Computer Vision and Deep Learning expertise to develop systems for automated extraction of features from remote sensor imagery, eventually leading to creation of high quality 3D synthetic environments. His research interests include fundamental and applied research covering the following areas: Feature Extraction & Classification from remote sensor imagery, 3D reconstruction of Synthetic environments, and Interpretability in Deep Neural Networks.

Hermann Brassard is a solution architect at Presagis Inc. Canada, Modeling and simulation Division. He holds an engineering degree in automation engineering and has 25 years of experience in the modeling and simulation industry. At Presagis, Hermann uses his wide knowledge of simulation and visualization in part to guide AI/ML research activities in areas of readily applicable components to serve rapid 3D world reconstruction.

Bhakti Patel is an Automation Developer at Presagis Inc. Canada, Modeling and Simulation Division. She holds a degree in Computer Science and has been working at Presagis since 2015. During the course of her professional career, Bhakti has developed a passion for automation and has been leveraging this to develop tools to facilitate GIS data processing. Her curiosity in combining different aspects of automation has allowed her to develop a pipeline to automate the generation of labelled datasets. In collaboration with the AI team at Presagis, Bhakti has been using her expertise to create large synthetic datasets to assist in machine learning.

Creating Geo-specific Synthetic Environments using Deep learning and Process Automation

Bodhiswatta Chatterjee, Hermann Brassard, Bhakti Patel
Presagis Inc. Canada
Montreal, Canada
Bodhiswatta.Chatterjee@presagis.com, Hermann.Brassard@presagis.com, Bhakti.Patel@presagis.com

INTRODUCTION

Creation of synthetic environments for training requires a large amount of Geographic Information System (GIS) data in the form of vectors such as building footprints, road networks, vegetation scatter, hydrography, etc. Publicly-available GIS information (e.g. Open Street Maps [OSM]) often contains insufficient amount of information and is not correlated with the Electro Optical (EO) or Infrared (IR) imagery of the area. Manually-labeled data of much better quality can be acquired, albeit at a higher cost.

Current advances in computer vision tasks allow object detection and semantic segmentation with relatively high accuracy using deep neural networks. These are ideal for the purpose of extraction of features like (building, roads, trees, water, etc.) from remote sensor imagery. A simple conversion of the extracted features into vectors will suffice to feed in a 3D synthetic environments reconstruction.

The extracted features can be used directly to create a geo-typical synthetic environment. However, creating geo-specific synthetic environments requires a large amount of information in the form of attributes for each extracted vector (e.g. building roof type, building height, road type, number of lanes on road, etc.). Those additional attributes can be acquired by combining manual labeling processes and a Digital Surface Model (DSM) of the area. Both of these are expensive and access to high resolution DSM to support building height assessment is often challenging.

Recently, there has been increasing interest in the extraction of geospatial features such as building footprints, road center lines, etc. from remote sensor imagery. There a few interesting and high quality datasets [INRIA (Maggiori et al, 2017), AIRS (Chen et al, 2018), SPACENET (Van et al, 2018), etc.] openly available for the purpose of training of deep neural networks for building segmentation, building footprint extraction or road network extraction. However, most of these datasets do not have attributes for important labels in support of 3D reconstruction such as roof type or building height. Since current deep learning techniques rely heavily on huge, labeled, datasets for training the networks, this presents a road block for the use of AI techniques to extract attributes-rich features from satellite imagery.

The contribution of this paper is twofold. First, we show how a pipeline of multiple neural-network-based techniques can be used to extract features efficiently from remote sensor imagery and attribute them in a multi-step process. We work with a cross-section of the entire problem (i.e. the building feature class which is considered as the most important feature for creation of urban synthetic environments). We use state-of-the-art deep neural network ICT-Net (Chatterjee et al, 2019) for extraction of building footprints as the first step of the pipeline.

Second, we address the most common problem for computer vision tasks: creation of huge labeled training datasets required to train deep neural network models. We tackle this problem by using two different approaches. The first approach relies on a state-of-the-art few-shot classification technique (Siamese networks) while the other, a much easier and more intuitive approach, relies on the generation of synthetic data to create a huge ML training and testing labeled dataset using our process automation software (Velocity). Velocity is an automated pipeline that allows large scale processing of GIS data and creation of procedural components to produce 3D terrain. For this paper we work with the most important building attribute after the footprint in support of 3D reconstruction: the roof types. This approach can still be extended to multiple attributes for any feature type. We also provide a performance-based comparative analysis of the two approaches.

The second stage of our pipeline (i.e. extraction of attributed feature vectors) helps us advance one step closer to creating geo-specific synthetic environments. In the end, we use existing automation processes to reconstruct the 3D environment using the produced attributed vectors. We create two synthetic environments, the first being geo-typical using manually labeled building footprints and the second being close to geo-specific using attributed building footprints extracted by the AI pipeline. We do a qualitative comparison of the two resulting environments.

METHODOLOGY

We use a multi-stage pipeline to extract features along with attributes from remote sensor imagery and create a geo-specific environment using process automation. The first stage of the pipeline takes in orthorectified RGB imagery as input and produces a binary (building/non-building) classification map. Next, it is further processed to extract the building boundaries as building footprints from the classification map. The next stage of the pipeline branches into two different approaches for extraction of the attribute roof type for each extracted building footprint. In the last stage of the pipeline we use process automation software to create a 3D synthetic environment by extruding the attributed building footprints. Figure 1 summarizes the AI pipeline for extraction of attributed building footprints.

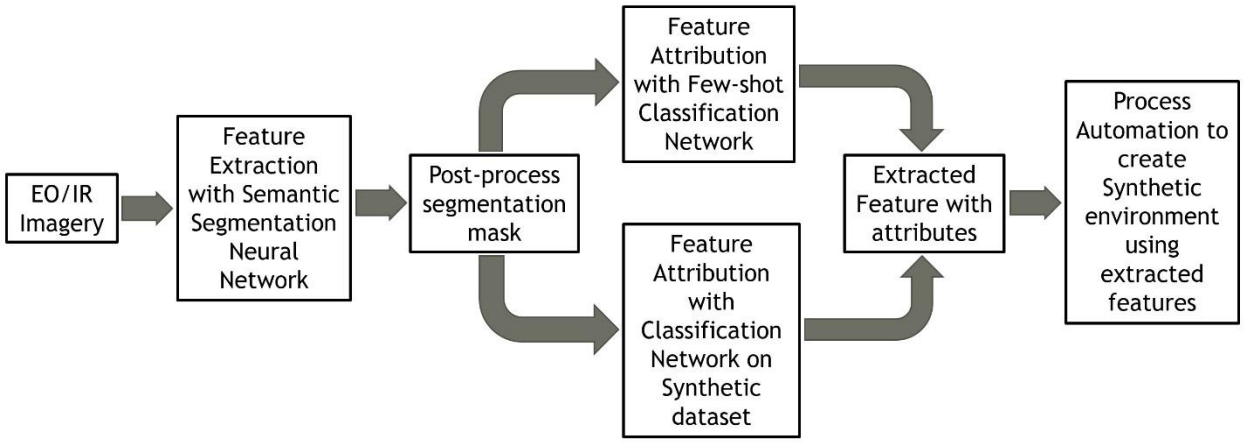

Figure 1. The diagram summarizes a pipeline for the work presented in this paper. The multi stage pipeline focuses on extraction of Features (i.e. Building footprints) and investigate the use of two different approaches (few-shot classification and synthetic data) for extraction of feature attributes such as roof type.

Building Footprint Extraction

The first stage of the pipeline is tasked with one of the most challenging tasks of computer vision: Semantic Segmentation i.e. per pixel classification of an image into different classes, in this case building/non-building classes. To successfully complete this task we use one of our recently developed technique ICT-Net (Chatterjee et al, 2019) a state-of-the-art deep neural network for semantic segmentation of buildings from satellite/aerial imagery. This technique has been demonstrated on two publicly available datasets for building segmentation (INRIA and AIRS). ICT-Net is an encoder-decoder style fully convolutional neural network with 103 convolutional layers. It uses feature-recalibrated dense blocks of symmetric but varying sizes in both the encoder and decoder parts of the architecture. The network was trained on the INRIA image labeling dataset (Maggiori et al, 2017) which consists of aerial imagery from 10 different cities in North America and Europe.

The output of the network is a pixel level mask of the same size as the input image with building pixels marked with 1 (white) and rest of the pixels are marked 0 (black). The next step towards extraction of the building footprints from the mask is to apply a well thought-out set of post processing steps to achieve refined, smooth and high quality building boundaries in the form of vectors. The post-processing includes a selection of one or more of the following techniques

(i) bounding box replacement, (ii) square simplification or (iii) Douglas-Peucker algorithm for polygon simplification, based on a threshold value.

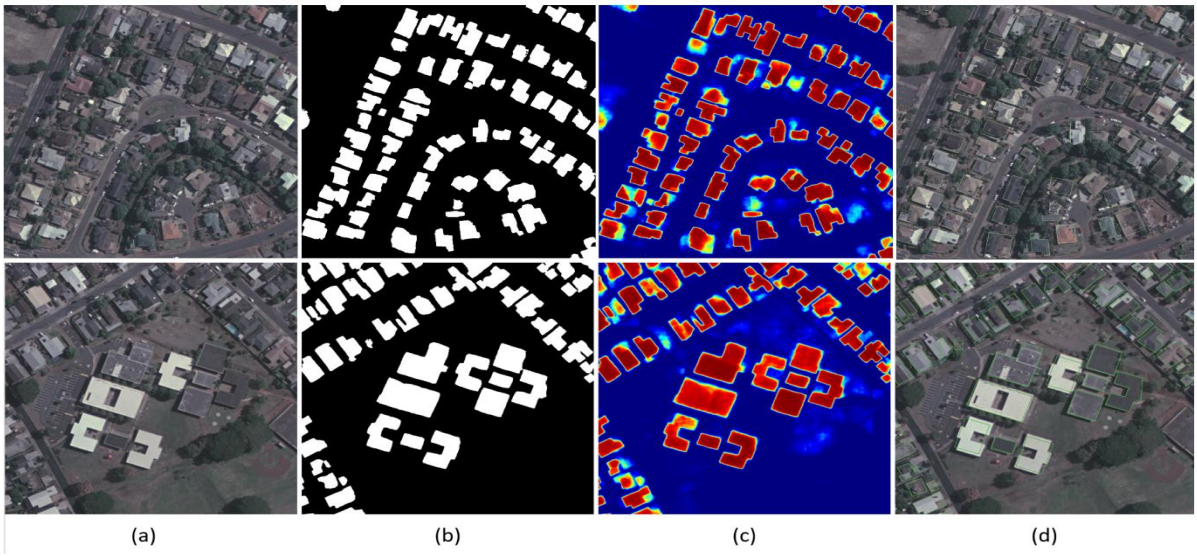

Figure 2. Shows results of building footprint extraction using ICT-Net on Hawaii imagery. (a) Color imagery. (b) Single channel prediction mask of same size as input imagery obtained as output of the network. (c) Confidence of network prediction shown as heat map. Red to Blue signifies high to low confidence for the building class. (d) Extracted and refined building footprints shown as green polygons on top of the imagery.

Roof Type Classification

The next stage of the pipeline is designed to extract properties (attributes) of the previously extracted building footprints. To demonstrate our technique in this paper we work with the most visually significant attributes (i.e. roof type) to create geo-specific synthetic environments but the same approach can be extended to other attributes. Current known supervised machine learning techniques work very well only if a large training dataset is available. To the best of our knowledge there were no existing datasets available for building a roof type classification system using supervised machine learning. Unavailability of labeled training data is a very common problem and current computer vision research community proposes two alternatives in such cases (i) Use a synthetic dataset for training of a deep convolutional neural network (CNN) or (ii) Use a few-shot classification technique. In this paper we do a quantitative comparison of both approaches to our roof type classification problem.

Synthetic Dataset Generation

Synthetic datasets are very commonly used for solving generic computer vision problems such as urban scene classification or autonomous driving when large labeled datasets are not available [e.g. Synthia (Ros et al, 2016), SunCG (Song et al, 2017), etc.]. In contrast, computer vision on remote sensing data is just starting to catch up with the use of synthetic data for training of machine learning models [e.g. (Krump et al, 2019)]. In order to generate the synthetic data used in our training, we have leveraged the existing 3D generation pipeline from Presagis along with 3D rendering capability. Using the software suite, over 1 million images were generated, allowing for a wide range of variety in terms of ground imagery, rooftops and rooftop shapes.

For this dataset we used satellite imagery with 30 cm Ground Sampling Density (GSD). First we extracted the geo-specific footprints of the buildings which match the exact size and shape of the buildings found in the imagery. Next we extruded a 3D building with a predetermined set of templates to generate a permutation of rooftops, each with different materials, colors and types of roof. After generating the buildings with different rooftops, we used a 3D rendering software (Vega Prime) to visualize the building on top of the satellite imagery.

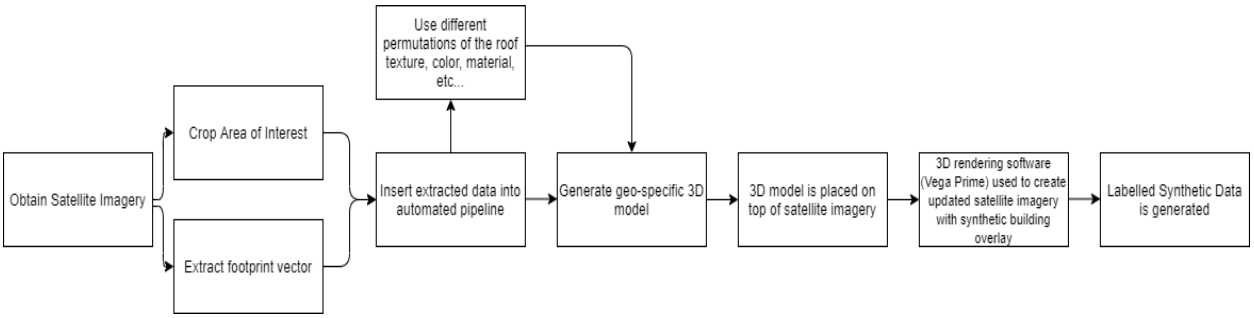

Figure 3. The diagram summarizes a workflow for generation of synthetic roof type dataset.

Finally, we placed the observer in Nadir view and calculated the field of view to match the resolution of the imagery. Using an automated process we took a snapshot of the rooftop with different variations and this process helped us produce different permutations of roof types using the same building. We also varied the sun position and illumination at different times of day to obtain variation in color and shadow of the building. Produced snapshots were saved and automatically labelled with the type, color and material of the roof type using the name of the saved file.

Few-shot classification

Solving computer vision tasks like image classification and object detection using minimal amount of labeled data has recently been an active area of research. There are many proposed techniques (Metric learning, Bayesian methods, Meta learning, etc.) which try to solve the same problem from different perspectives. To date, the current accuracies of these techniques are not comparable to supervised learning techniques. The only exception is the use of Siamese networks (Taigman et al, 2014) – a metric learning based approach which is currently the state-of-the-art for Face detection systems and results in very high level of accuracy. It is also used in the domain of optical character recognition (Koch et al, 2018). To the best of our knowledge this is the first time Siamese networks are being introduced in the context of image classification in remote sensing domain.

The principle behind Siamese networks is a CNN-based architecture which learns a distance metric between two embedding of two input images passed through the same network (with same weights). The comparison of the two images in the latent space produces a distance score that is used to estimate the class membership of each image. The training of such networks requires the design of a special loss function call contrastive (triplet) loss.

Synthetic Environment Generation

To generate the synthetic database, we use the Velocity framework. Velocity is an automated pipeline that allows us to process GIS data on a large scale and create procedural databases. Velocity is composed of what we call "operators" – which are sub-processes that, when combined, create an automated workflow. The combination of these operators result in a "recipe".

We use the recipe to manipulate, attribute and process the GIS source data (which can consist of vectors, imagery and raster data). An attribution such as height is inferred using algorithms that are based on the area of the footprint. If no additional information is given in the pipeline, the roof type and color are randomly attributed to the buildings from a library of building templates, resulting in a variety of 3D models in the database. However, with the attribution from the AI pipeline (roof type and roof color), it will be processed in the recipe by assigning specific building templates to the footprint according to the information given; this process constructs accurate models for the AOI (area of interest). Within the AI pipeline the roof color is extracted as the average color value of the area predicted as a single building, and returned as an attribute for each building. The color of the roof type is selected within the recipe using this attribute and this provides a better representation of the real world environment once Velocity takes the processed data and starts publishing the new content database (i.e. the 3D scene).

EXPERIMENTS AND RESULTS

To validate the proposed use of synthetic data for training of neural networks, we devised two sets of experiments. The first set comprised of three state-of-the-art image classification techniques trained on the generated synthetic roof dataset. For the second set, we trained a Siamese neural network, very similar to what was proposed in (Koch et al, 2018) on a small subset of a synthetically-labeled roof type training dataset.

Data Preparation for Training

The synthetically generated dataset consists of 1.1 million images built on top of 200 selected footprints with a crop of size 256x256 pixels. The dataset was subdivided into 3 parts training, validation and test sets with approximate proportions of 80%, 10% and 10% respectively without any overlaps. All images of 160 randomly selected footprints were used as the training dataset and the rest of the images are split equally to form validation and test datasets. For all 3 supervised classification techniques we used the complete training dataset along with a number of standard data augmentation techniques [rotation (up to 10 degrees), flip (horizontal/vertical), adaptive contrast and brightness alterations, etc.]. For the purpose of training of the Siamese network we took a subset of the training dataset 1760 images (i.e. 160 x 11 images) equally distributed in terms of roof types (i.e. 11 roof types for each footprint). For the purpose of reporting, the validation and test set remained consistent for all 4 network evaluation techniques.

Training Process for Supervised Classification Networks

The selected image classification networks have demonstrated very good results on ImageNet (Deng et al, 2009), one of the most recognized dataset in computer vision research for image classification. The first of the selected networks is a specific version of residual networks (He et al, 2016) ResNet50 which consists of 50 convolutional layers organized into 5 residual blocks. Next we used a densely connected convolutional neural network DenseNet121 (Huang et al, 2017) which contained 121 convolutional layers organized into 4 dense blocks with a growth rate (k) of 32. Although this network has many more convolutional layers, the number of trainable parameters is only 7 million as compared to 23 million for ResNet50 due to the efficient design of dense blocks in DenseNet. A lower number of trainable parameters in a deep neural network helps the network to learn the most important features and, in-turn, to generalize to perform well on the test dataset. The last selected network is called MobileNet (Howard et al, 2017), a unique neural network architecture designed using two special concepts of depthwise convolution and pointwise convolution built into 1 block. This network is made up of 5 such blocks, consisting of only 2.2 million trainable parameters in total and was able to get state-of-the-art on the ImageNet dataset.

For training of all the neural networks we used a common set of hyper parameters. The input to the network was 256x256 pixels in size with 3 channel and the numerical values normalized between 0 and 1. All networks used cross-entropy loss with Adam optimizer with an initial learning rate of 0.01. All the networks used a learning rate decay with a factor of 0.5 when the validation loss plateaus for 3 epochs and an early stopping when the validation loss does not decrease for 10 epochs. We used a standard set of data augmentation techniques to further increase the size of dataset and to reduce overfitting of network parameters.

Training Process for Siamese Network

The term Siamese refers to twins. This category of architectures contain two streams of convolutional neural networks running in parallel; they are not different networks but are two copies of the same network sharing the same parameters, hence the name Siamese Networks. The two input images (x1 and x2) are passed through the convolutional layers to generate a fixed length feature vector for each (h(x1) and h(x2)). After the neural network model is trained properly, we can make the following hypothesis: If the two input images belong to the same character, then their feature vectors must also be similar, while if the two input images belong to the different characters, then their feature vectors will also be different. Thus the element-wise absolute difference between the two feature vectors must be very different in both cases above. The similarity score generated by the output sigmoid layer must also be different in these two cases.

The model consists of two streams of 13 convolution layers organized similarly to the convolutional layers in VGG16 (Simonyan et al, 2014) followed by two global average pooling layers. These two streams merge into an absolute difference layer which is used to replace the contrastive loss at the end. It is followed by a fully connected sigmoid layer. The network uses binary cross-entropy loss and Adam optimizer with an initial learning rate of 0.0001. The input to the network is a pair of 256x256 color images with their numerical values normalized between 0 and 1. One of the images acts as the anchor and the other is either of the same or different class (roof type) chosen at random. This network also uses a learning rate decay with a factor of 0.5 when the validation loss plateaus for 3 epochs and an early stopping when the validation loss does not decrease for 10 epochs. At the time of training we selected a subset of only 660 images from the validation dataset, equally distributed in term of roof types to evaluate the validation loss. In the end, we reported our evaluation in the results section on the complete validation dataset.

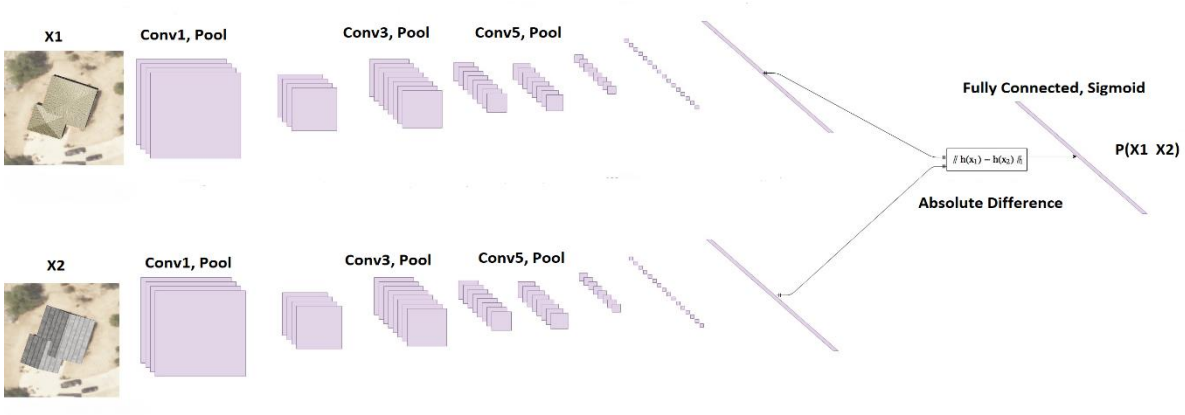

Figure 4. This diagram illustrates a simplified version of the Siamese neural network used in this paper. The number of Convolutional layers used in the network varies from 7 in the image to 13 for our custom Siamese network.

Results

As mentioned previously, we split the dataset into three non-overlapping parts for training, validation and test. For all the models we used only the training subset to train our neural networks and reported the accuracy (in percentage) on the validation subset as well as test subset. All three supervised neural networks performed well as compared to the Siamese approach, which showed very poor performance on this task. *MobileNet was able to achieve the best performance of 77.21% on validation and 77.01 on the test dataset.* It is very difficult to access the exact reason for poor performance of Siamese networks on this task as many recent works (Kihyuk Sohn, 2016, Roy et al, 2019) have demonstrated that training of Siamese networks possess additional challenges other than neural network architecture design and sampling of data plays a crucial role in this process. Metric learning based techniques like Siamese networks are active and rapidly developing areas of machine learning research but this experiment clearly demonstrates the challenges with this technique. It is also evident that use of classification networks with large amount of synthetically generated data is a much suitable alternative, when sufficient training data is not available. Table 1 shows a quantitative comparison of the 4 discussed techniques on validation and test datasets.

Table 1. Quantitative comparison of the discussed techniques on Validation and Test datasets

Technique	Validation Accuracy (%)	Test Accuracy (%)
ResNet50	71.44	69.02
DenseNet121	65.79	57.19
MobileNet	**77.21**	**77.01**
Siamese Network	9.09	9.09

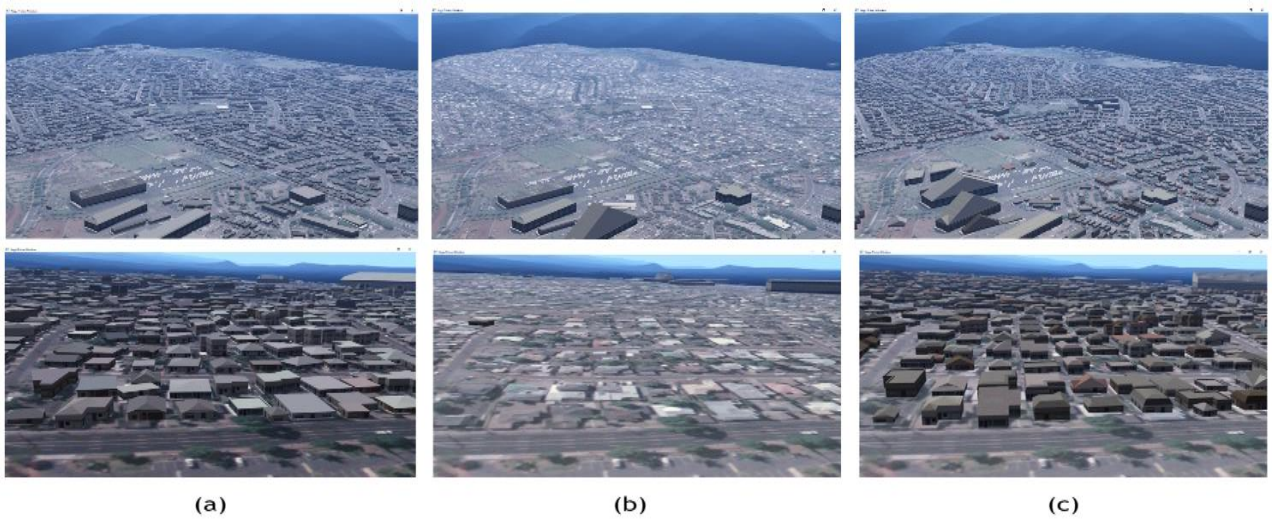

Figure 5. Shows 3D reconstruction of the same area in Hawaii using 3 different techniques. (a) Attributed building footprints extracted using the AI pipeline. (b) OSM data from Hawaii and (c) NGA freely available vector data for Hawaii.

We also show a qualitative comparison of the three different 3D reconstruction of the same area of Hawaii using vectors extracted from AI pipeline, OSM data for Hawaii and NGA freely available data for Hawaii in Figure 5. The figure (a) clearly shows that the AI pipeline is able to extract significant amounts of information from the imagery in the form of Building footprints and its attributes such as roof type and roof color. Figure (b) shows the same area reconstructed using OSM data, which is the most common freely available data; due to limited coverage there are very few buildings in the figure. Figure (c) shows a 3D reconstruction of the same area using freely-available building footprint vectors from the office of planning, State of Hawai'i (NGA), In some cases the building boundaries are sharper as they are manually labeled, which requires significant manual effort. Also, NGA vector data did not include any attributes so the roof type and color were assigned randomly. Table 2 summarizes the different attributes of the vector data used for the creation of 3 different 3D models. AI extraction refers to the vector data produced by our proposed pipeline whereas Open Street Map and NGA refers to vector data made available by the corresponding agencies.

Table 2. This table shows a comparison of the attributes of the vector data used for creation of the 3D models

Vector source	**AI extraction**	**Open Street Map**	**NGA**
Footprint	AI produced from imagery	Manual – from crowd source	Manual – from Gov agency
Building type attribution	Not extracted	Manual – from crowd source	Manual – from Gov agency
Roof type attribute	Extracted from imagery	Not available	Not available

CONCLUSION

In this paper we presented a multi-level Deep Learning-based AI pipeline for the extraction of attributed building footprints with attributes such as roof type and roof color for each extracted building. Using the extracted and derived attributes such as the area of the footprint, the building heights were estimated. This technique replaces the acquisition of expensive and difficult-to-procure high-resolution DSM data for the creation of geo-specific synthetic environments. We also demonstrated the use of synthetic data for training of Deep Learning models in the remote sensing domain to solve the problem of roof type classification where no labeled training dataset exists. A performance-based comparative analysis of multiple classification techniques on the synthetic data was also

performed to provide insights on how synthetic/hybrid datasets can be used when labeled training datasets are not available.

Furthermore, we have shown qualitative results from the AI pipeline in the form of 3D reconstruction of an area of interest on the island of Hawaii. Visual comparison of the different 3D models demonstrates the advantage of using AI (computer vision) for extraction of attributed features from imagery for the purpose of 3D scene reconstruction. In the future we would like to extend our work to improve accuracy of roof type classification as well as building footprint extraction, especially in cases of partially observable buildings (e.g. when part of a building is hidden by trees).We would also like to extend the AI-based approach for extraction of other attributed feature classes such as urban vegetation and roads.

ACKNOWLEDGEMENTS

The authors would like to express their appreciation for the initial discussion and insights provided by Prof. Charalambos Poullis, Immersive and Creative Technologies Lab, Concordia University, Montreal, and for all the support provided by the Presagis team.

REFERENCES

Maggiori, E., Tarabalka, Y., Charpiat, G., & Alliez, P. (2017, July). Can semantic labeling methods generalize to any city? The inria aerial image labeling benchmark. In 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS) (pp. 3226-3229). IEEE.

Chatterjee, B., & Poullis, C. (2019, May). On building classification from remote sensor imagery using deep neural networks and the relation between classification and reconstruction accuracy using border localization as proxy. In 2019 16th Conference on Computer and Robot Vision (CRV) (pp. 41-48). IEEE.

Chen, Q., Wang, L., Wu, Y., Wu, G., Guo, Z., & Waslander, S. L. (2018). Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings. arXiv preprint arXiv:1807.09532.

Van Etten, A., Lindenbaum, D., & Bacastow, T. M. (2018). Spacenet: A remote sensing dataset and challenge series. arXiv preprint arXiv:1807.01232.

He, K., Zhang, X., Ren, S., & Sun, J. (2016, October). Identity mappings in deep residual networks. In European conference on computer vision (pp. 630-645). Springer, Cham.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).

Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7132-7141).

Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1701-1708).

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). IEEE.

Koch, G., Zemel, R., & Salakhutdinov, R. (2015, July). Siamese neural networks for one-shot image recognition. In ICML deep learning workshop (Vol. 2).

Ros, G., Sellart, L., Materzynska, J., Vazquez, D., & Lopez, A. M. (2016). The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3234-3243).

Song, S., Yu, F., Zeng, A., Chang, A. X., Savva, M., & Funkhouser, T. (2017). Semantic scene completion from a single depth image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1746-1754).

Krump, M., Ruß, M., & Stütz, P. (2019, October). Deep Learning Algorithms for Vehicle Detection on UAV Platforms: First Investigations on the Effects of Synthetic Training. In International Conference on Modelling and Simulation for Autonomous System (pp. 50-70). Springer, Cham.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Sohn, K. (2016). Improved deep metric learning with multi-class n-pair loss objective. In *Advances in neural information processing systems* (pp. 1857-1865).

Roy, S. K., Harandi, M., Nock, R., & Hartley, R. (2019). Siamese networks: The tale of two manifolds. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 3046-3055).