

Deep Learning-Based Improved Automatic Building Extraction from Open-Source High Resolution Unmanned Aerial Vehicle (UAV) Imagery



Chintan B. Maniyar and Minakshi Kumar

Abstract Automatically extracting buildings from remotely sensed imagery has always been a challenging task, given the spectral homogeneity of buildings with non-building features as well as the complex structural diversity within the image. Traditional machine learning (ML) based methods deeply rely on a huge number of samples and are best suited for medium-resolution images. Unmanned aerial vehicle (UAV) imagery offers the distinct advantage of very high spatial resolution, which is helpful in improving building extraction by characterizing patterns and structures. However, with increased finer details, the number of images also increases many folds in a UAV dataset, which require robust processing algorithms. Deep learning algorithms, specifically Fully Convolutional Networks (FCNs) have greatly improved the results of building extraction from such high resolution remotely sensed imagery, as compared to traditional methods. This study proposes a deep learning-based segmentation approach to extract buildings by transferring the learning of a deep Residual Network (ResNet) to the segmentation-based FCN U-Net. This combined dense architecture of ResNet and U-Net (Res-U-Net) is trained and tested for building extraction on the open source Inria Aerial Image Labelling (IAIL) dataset. This dataset contains 360 orthorectified images with a tile size of 1500 m² each, at 30 cm spatial resolution with red, green and blue bands; while covering total area of 805 km² in select US and Austrian cities. Quantitative assessments show that the proposed methodology outperforms the current deep learning-based building extraction methods. When compared with a singular U-Net model for building extraction for the IAIL dataset, the proposed Res-U-Net model improves the overall accuracy from 92.85% to 96.5%, the mean F1-score from 0.83 to 0.88 and the mean IoU metric from 0.71 to 0.80. Results show that such a combination of two deep learning architectures greatly improves the building extraction accuracy as compared to a singular architecture.

C. B. Maniyar (✉) · M. Kumar

Photogrammetry and Remote Sensing Department, Indian Institute of Remote Sensing, Dehradun UK 248001, India

e-mail: chintanmaniyar@gmail.com

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023

K. Jain et al. (eds.), *Proceedings of UASG 2021: Wings 4 Sustainability*,

Lecture Notes in Civil Engineering 304,

https://doi.org/10.1007/978-3-031-19309-5_5

Keywords Transfer learning · Fully convolutional networks · Image segmentation · Building extraction

1 Introduction

1.1 Background

Remote sensing imagery, both satellite and aerial, contains a lot of terrain-feature-specific information such as land-cover spread, building footprints, waterbody extent, vegetation and forest boundaries etc. Extracting this feature information without losing relative context within the image is a very important remote sensing image processing milieu [1, 2]. Feature extraction is usually done by identifying a common pattern among pixels and grouping them together, that group of pixels then being a feature [3]. One of the most crucial aspects for accurate image feature extraction is finer spatial details such as edges and corners. Primitive feature extraction methods were time-consuming and required a lot of expensive human intervention [4]. This was mostly because of the unavailability of higher spatial resolution data in conjunction with the technical infrastructure at the time. However, with advancements in digital systems for image processing and also the increased availability and accessibility of high spatial resolution data from both satellites and Unmanned Aerial Vehicles (UAVs), image feature extraction has consistently been one of the hottest research topics in remote sensing image processing [5].

1.2 Previous Works

In remote sensing feature extraction, building extraction is one of the most vital aspects of research. With its applications spread in various pipelines of urban mapping and management, disaster management, change detection, maintaining and updating geodatabases etc., building extraction has caught the attention of researchers worldwide for developing robust and accurate algorithms to automate the process [6]. Primitive methods of building extraction were based on applying statistical and morphological operations on individual pixels to group them together [7], hence automating the task up to some extent. One of the most prevalent issues in building extraction that has propagated from early methods to the recent methods is the differentiation of foreground and background as well as building and non-building objects [8]. To be able to differentiate between these, spectral and geometrical cues such as color, shape and line have been used to extract buildings from very high-resolution imagery [9]. Another study combined distinctive corners while estimating building outlines to extract buildings [10], but was unable to extract irregular-shaped buildings. In the beginning of the decade, a generic index called Morphological Building Index (MBI) was introduced to extract buildings from high-resolution satellite imagery, based on

spectral information [11]. While this method was able to successfully extract buildings with an irregular shape, it failed in shadowy regions and also could not extract buildings located close by (instance extraction). A consequent study to MBI proposed a Morphological Building/Shadow Index which defined a building index as well as a shadow index, and was specifically aimed at bridging the shortcomings of the MBI method [12].

With the recent availability of strong computing systems as well as finer resolution data, artificial intelligence-based deep learning algorithms such as Convolutional Neural Networks (CNNs) are being aggressively used for building extraction given their advantage of hierarchical feature extraction without losing any contextual information [13, 14, 15]. In general, a deep learning architecture consists of a network structure with many hidden layers leading to hierarchical feature extraction thus, eliminating the problem of inadequate representation of learning features [16]. Building-A-Nets is an adversarial network to for robust extraction of building rooftops. Multiple Feature Reuse Network (MERN) is a resource-efficient rich CNN to detect building edges from high spatial resolution satellite imagery [17]. A special type of pre-trained CNN, called a Fully Convolutional Network (FCN) is also being widely used for transfer learning-based building extraction. A few such popular FCNs are VGG-16 [18], ResNet [19], Deeplab [20], DenseNet [21], SegNet [22] and U-Net [23]. Studies specifically on building extraction from UAV images have also increased of late. SegNet and U-Net have been used in an ensemble manner to improve building footprint extraction from high-resolution UAV imagery [24]. Techniques such as dilated spatial pyramid pooling [25], multi-stage multi-task learning [26], and channel attention mechanisms [27] have been used to improve the building segmentation accuracy from UAV data. Variants of U-Net architecture have also been tested for building extraction and studies indicate that the U-Net is the most suitable for dense image building extraction [15, 28, 29].

1.3 Objective and Summary

Sometimes, the FCN based segmentation is visually degraded in case of blurred building boundaries [30]. Moreover, high spatial resolution data is generally restricted to three or four spectral channels, which makes it difficult to differentiate buildings and other spatially similar features [24]. To address these issues, this study proposes a deep learning-based segmentation approach that combines a pre-trained FCN with a U-Net being trained for building extraction, to extract buildings from high-resolution RGB UAV imagery. The learning of a deep Residual Network (ResNet) trained on the ImageNet dataset is transferred to the segmentation-based FCN U-Net, hence forming a combined Res-U-Net architecture. In this Res-U-Net, the pre-trained ResNet helps capture more context in case of features spatially similar to buildings while the U-Net learns building segmentation based on a unique loss function (discussed in Sect. 2.3) that simultaneously accounts for crispness as well as the region of a segmented building, hence preventing prediction leakage

outside of feature in case of blurred boundaries. Consequent sections of the paper discuss the dataset details, data preparation and training methodology, results and their inferences, and conclude the study.

2 Dataset Details

This study uses the Inria Aerial Image Labelling (IAIL) dataset. This dataset contains a total of 360 orthorectified images (180 for training and 180 for testing) with a tile size of 1500 m² each, at 30 cm spatial resolution with red, green and blue bands. Each image is of size 5000 × 5000 pixels. While covering an area of 81 km²/city in select 3 US cities of Austin, Chicago, Kitsap County and select Austrian cities of Vienna and West Tyrol, this dataset contains 36 images from each city having high variance in terms of urban density and building spacing. Moreover, numerous instances of shadowy features and shadowy backgrounds are present, especially in the images from Chicago, US. The ground truth of the training set is provided as a binary feature image with only two classes namely building and non-building. Since ground truth is provided only for the training set of 180 images, we use only those 180 images to train and validate our model. Figure 1 shows the UAV image and its corresponding ground truth as available from the IAIL training set, for each of the five cities.

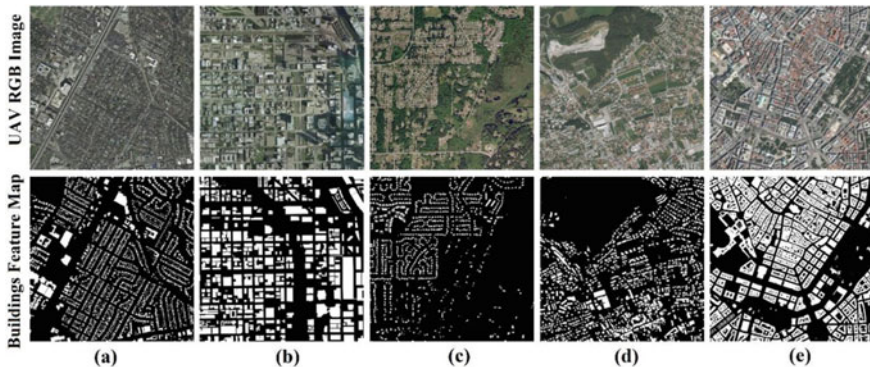


Fig. 1 Data samples from the IAIL dataset, one from each city **a** Austin, USA, **b** Chicago, USA, **c** Kitsap County, USA, **d** West Tyrol, Austria, **e** Vienna, Austria

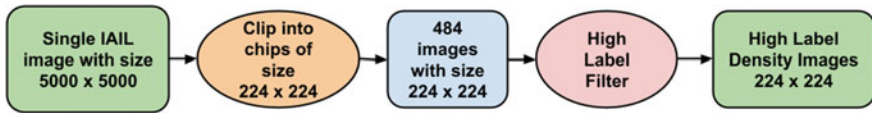


Fig. 2 Data preparation methodology for a single image

3 Methodology

3.1 Data Preparation Methodology

A single image is of size 5000×5000 pixels. We further split it into small data chips of size 224×224 pixels in accordance to the proposed network architecture. This results into 484 such tiles from a single image. However, certain number of chips contain no buildings or hardly any buildings at all, creating a bias in the type of data which could result in model misfit. To ensure uniformity of 224×224 chips in terms of buildings, we further filter the 484 chips using a High Label Filter (Eq. 1). This is basically a ratio of the number of labelled pixels to the total number of pixels in a 224×224 chip. We use a threshold of 0.3 in the High Label Filter to further filter these 484 chips. This excludes the chips having label density less than 30% and hence the earlier bias in the data is now removed. Figure 2 shows the data preparation methodology for a single image. This process is performed for all 180 images as well as labels. Passing the 87,120 224×224 chips obtained from 5000×5000 180 images (180×484) through the High Label Filter, we get 27,164 224×224 chips. The proposed model is trained and validated on these 27,164 chips and entire images of size 5000×5000 are used for testing.

$$HLF = \frac{\sum_{i=0}^{224 \times 224} \text{building_pixel}_i}{\sum_{i=0}^{224 \times 224} \text{image_pixel}_i} \quad (1)$$

3.2 Network Architecture

In this study, the U-Net architecture is implemented with a dynamic decoder to learn building extraction as a fully convolutional network (FCN). The whole architecture essentially consists of two major operations—image contraction performed by the encoder and image expansion performed by the decoder (Fig. 3). The encoder is responsible for pooling out the necessary information from within the convolution kernel which is done by max pooling operations. The decoder helps preserve precise local information such as building edges in case of blurred images which is done by upsampling and convoluting over transposed kernels. Each step of encoder is

Table 1 Specific proposed network architecture with individual layer parameters

Layer			Kernel size	Output shape	Stride
Conv2d			7×7	$64 \times 112 \times 112$	2
Sequential block 1	Conv2d		3×3	$64 \times 56 \times 56$	1
	Conv2d		3×3	$64 \times 56 \times 56$	1
	Conv2d		3×3	$64 \times 56 \times 56$	1
Sequential block 2		Down block 1	1×1	$128 \times 28 \times 28$	2
	Conv2d		3×3	$128 \times 28 \times 28$	1
	Conv2d		3×3	$128 \times 28 \times 28$	1
	Conv2d		3×3	$128 \times 28 \times 28$	1
	Conv2d		3×3	$128 \times 28 \times 28$	1
Sequential block 3		Down block 2	1×1	$256 \times 14 \times 14$	2
	Conv2d		3×3	$256 \times 14 \times 14$	1
	Conv2d		3×3	$256 \times 14 \times 14$	1
	Conv2d		3×3	$256 \times 14 \times 14$	1
	Conv2d		3×3	$256 \times 14 \times 14$	1
	Conv2d		3×3	$256 \times 14 \times 14$	1
Sequential block 4		Down block 3	1×1	$512 \times 7 \times 7$	2
	Conv2d		3×3	$512 \times 7 \times 7$	1
	Conv2d		3×3	$512 \times 7 \times 7$	1
Conv2d			3×3	$1024 \times 7 \times 7$	1
Conv2d			3×3	$512 \times 7 \times 7$	1
	Conv2d		3×3	$1024 \times 7 \times 7$	
U-Net block 1		Pixelshuffle		$256 \times 14 \times 14$	1
	Conv2d		3×3	$512 \times 14 \times 14$	1
	Conv2d		3×3	$512 \times 14 \times 14$	1
	Conv2d		3×3	$1024 \times 14 \times 14$	
U-Net block 2		Pixelshuffle		$265 \times 28 \times 28$	1
	Conv2d		3×3	$384 \times 28 \times 28$	1
	Conv2d		3×3	$384 \times 28 \times 28$	1
	Conv2d		3×3	$768 \times 28 \times 28$	
U-Net block 3		Pixelshuffle		$192 \times 56 \times 56$	1
	Conv2d		3×3	$256 \times 56 \times 56$	1
	Conv2d		3×3	$256 \times 56 \times 56$	1
	Conv2d		3×3	$512 \times 56 \times 56$	
U-Net block 4		Pixelshuffle		$128 \times 112 \times 112$	1
	Conv2d		3×3	$96 \times 112 \times 112$	1
	Conv2d		3×3	$96 \times 112 \times 112$	1

(continued)

Table 1 (continued)

Layer			Kernel size	Output shape	Stride
PixelShuffle_ICNR	Conv2d		1×1	$384 \times 112 \times 112$	1
		Pixelshuffle		$96 \times 224 \times 224$	
Sequential extension	Conv2d		3×3	$99 \times 224 \times 224$	1
	Conv2d		3×3	$99 \times 224 \times 224$	1
Conv2d			1×1	$222 \times 224 \times 224$	1

corresponding labels. The network was trained with a batch size of 6 and a patch size of 224×224 for 30 epochs, with roughly 1200 batches being processed per epoch. The training was cut-off based on loss convergence (Fig. 5a). The learning was carried out on nearly 20 million parameters extracted at different layers of the network. The network was optimized with ADAM optimizer at a learning rate of 0.0001 and a decay rate of 0.9.

A unique combination of Binary Cross Entropy (BCE) loss (Eq. 2) and dice loss (Eq. 3) was used to train the network. BCE is a probability distribution-based loss [33] and hence was used to minimize the entropy between the prediction and the ground truth in terms of buildings as features. It was also helpful in preserving the crispness near the boundary regions. Dice loss is a region-based Intersection-over-Union like metric [34] and it was used to maximize the overlap and similarity between the

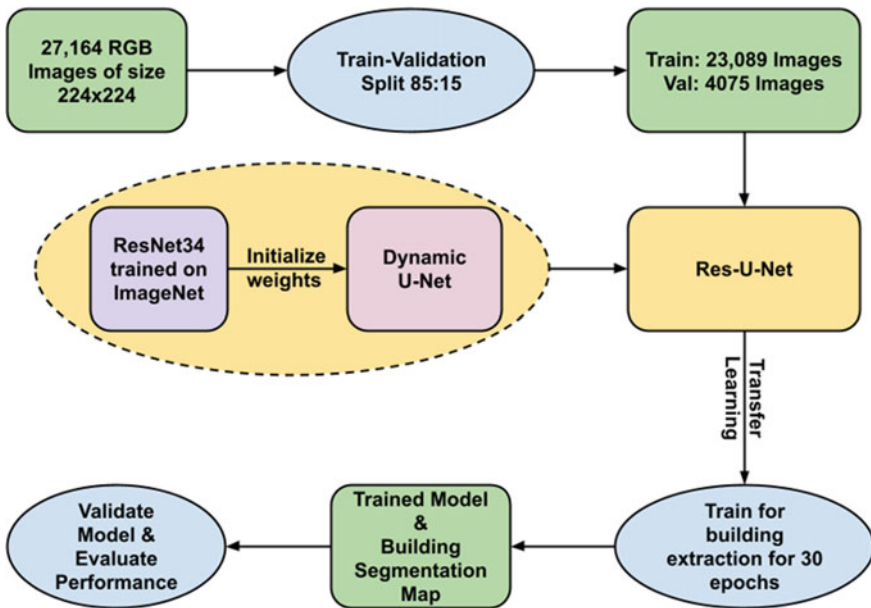


Fig. 4 Network training methodology for building extraction using transfer learning

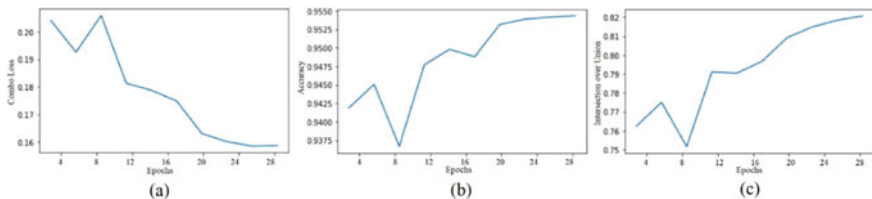


Fig. 5 a Combo loss variation, b accuracy variation and c IoU variation in 30 epochs of training

predicted region and the ground truth of the feature region. Hence, a combo loss was defined (Eq. 4) which focused on both boundary and region preservation. Figure 5a shows the loss-based convergence of the model after 30 epochs of training. After training for 30 epochs 7 and processing 36,000 batches the model began to converge and was saved at the end of 30 epochs with an overall accuracy of 95.7% and mean Intersection over Union (IoU) of 0.83.

$$BCE_{Loss} = -\frac{1}{patchsize} \sum_{i=1}^{patchsize} g_i \times \log p_i + (1 - g_i) \times \log(1 - p_i) \quad (2)$$

$$Dice\ Loss = \frac{2 \times \sum_{i=0}^{patchsize} p_i g_i}{\sum_{i=0}^{patchsize} p_i^2 + \sum_{i=0}^{patchsize} g_i^2} \quad (3)$$

$$Combo\ Loss = BCE_{Loss} + DiceLoss \quad (4)$$

where g = ground truth image, p = predicted building mask

4 Results and Discussion

Figure 6 shows the results for building extraction for select RGB images from each city of the IAIL dataset. The first column is the input to the model, the second column is the ground truth, the third column is the segmented building map as predicted by the model and the fourth column shows the evaluation of the prediction with True Positives (TP) in white, True Negatives (TN) in black, False Positives (FP) in red and False Negatives (FN) in yellow. These are original images of size 5000×5000 from the IAIL dataset. The predictions are obtained by clipping to chips of 224×224 , segmenting buildings and then again merging to the original size of 5000×5000 . In Fig. 6 we try to show all the different conditions for building extraction such as the surrounding land-cover classes, urban density, shadows etc. from each city. Figure 6a, c, f show successful building extraction in case of high urban density with closely spaced buildings, with rare instance segmentation challenges. Figure 6b shows effective building extraction even in shadowy regions. It can be noted that the shadows are not falsely classified as buildings, which has been a

very popular challenge in building extraction [12]. Figure 6a, b, f show successful building extraction in presence of spectrally similar features such as cemented roads and parking lots as well as spatially similar features such as roads, open grounds and vegetation patches having shape similar to buildings. The model is also able to segment buildings even when the dominant land cover in the image is not urban— Fig. 6d, e contain a large cover of vegetation, Fig. 6b, e contain a large area of water.

To quantify the prediction made by the model in terms of binary segmentation, the metrics of accuracy (4), precision (5), recall (6) and F1-score (7) were used. To further perform a feature-based evaluation, object-based metrics such as branching factor (8), miss factor (9), detection percentage (10) and IoU or quality percentage (11) (otherwise also popularly known as jaccard index) were used. Table 2 shows the metrics of the individual images in Fig. 6.

$$accuracy = \frac{tp + tn}{tp + tn + fp + fn} \quad (5)$$

$$precision = \frac{tp}{tp + fp} \quad (6)$$

$$recall = \frac{tp}{tp + fn} \quad (7)$$

$$f1 = 2 \times \frac{precision * recall}{precision + recall} \quad (8)$$

$$branchingFactor = \frac{fp}{tp} \quad (9)$$

$$missFactor = \frac{fn}{tp} \quad (10)$$

$$detectionPercentage = 100 \times \frac{tp}{tp + fn} \quad (11)$$

$$qualityPercentage/IoU = 100 \times \frac{tp}{tp + fn + fp} \quad (12)$$

where tp = True Positive, fp = False Positive, tn = True Negative and fn = False Negative.

Figure 7 shows the city-wise metrics of model validation. Tyrol West and Vienna from the IAIL dataset exhibit highly favourable conditions for building extraction. Extracting buildings from Chicago and Kitsap has been the most challenging. This is due to shadowy regions, typically the shadows being cast on other buildings. Though the proposed model successfully discriminates between shadowy regions and buildings and avoids shadows as false positives, it faces significant challenges

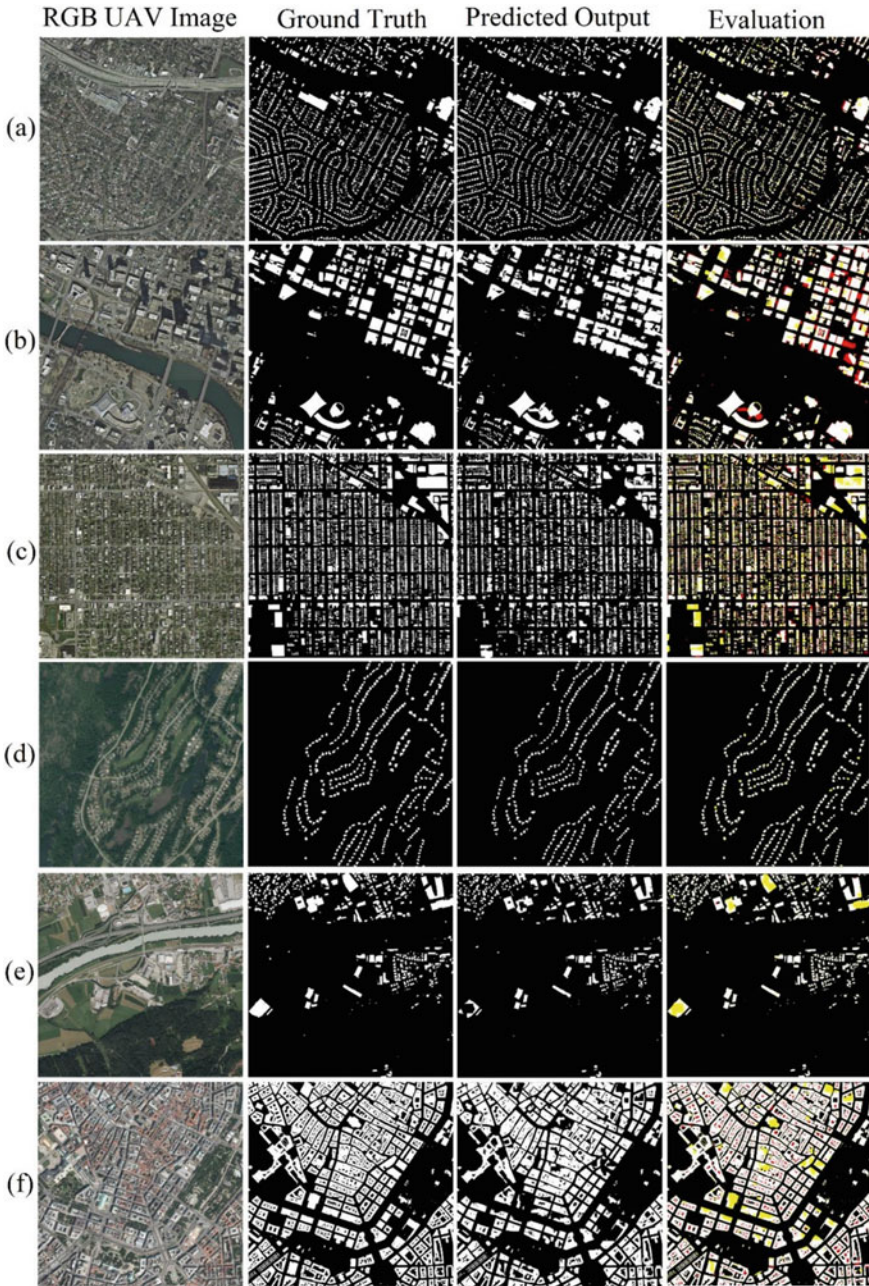


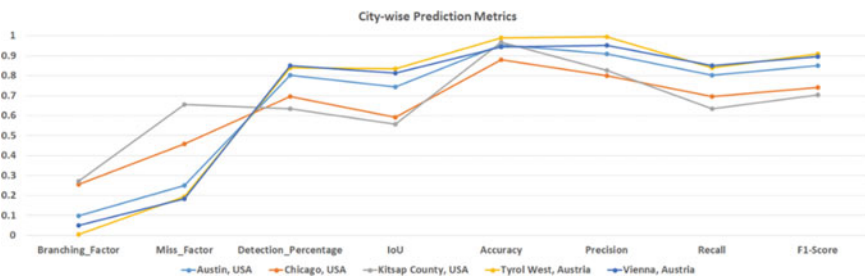
Fig. 6 Select instances of building extraction results from each city of the IAIL dataset. First column is RGB input to the model, the second column is model prediction for building segmentation, third column is ground truth and the fourth column is the evaluation image showing TP (white), TN (black), FP (red) and FN (yellow). **a, b** From Austin, USA, **c** from Chicago, USA, **d** from Kitsap County, USA, **e** from Tyrol West, Austria, **f** from Vienna Austria

Table 2 Metrics for individual images of Fig. 6

	Accuracy	Precision	Recall	f1-score	Branching factor	Miss factor	Detection percentage	IoU
Figure 6a	0.961	0.943	0.802	0.867	0.060	0.246	0.802	0.765
Figure 6b	0.942	0.850	0.764	0.859	0.177	0.152	0.868	0.752
Figure 6c	0.870	0.833	0.666	0.745	0.200	0.500	0.666	0.589
Figure 6d	0.991	0.997	0.851	0.920	0.001	0.176	0.851	0.845
Figure 6e	0.982	0.994	0.796	0.889	0.006	0.257	0.796	0.801
Figure 6f	0.927	0.938	0.893	0.915	0.006	0.120	0.893	0.843

in extracting the buildings which are under shadows. This drastically increases the rate of false negatives, as the model excludes the buildings under shadows as only shadowy regions (Fig. 8a, b). A potential reason for this could be loss of spectral variance as well as the spatial distinction of a building that is under shadow. Moreover, another isolated issue encountered in a Kitsap image is a patch of waterbody being falsely segmented as building, resulting into a high number of false positives (Fig. 8c). This could be due to multiple reasons such as spectral similarity of the waterbody area due to turbidity, or saturation of DN values in those areas due to direct glint on sensor. Such instances of shadowed buildings and typical water areas are prominent in the images from Chicago and Kitsap and hence the extraction results are lowest for these two cities from the IAIL dataset. Figure 8 shows select instances buildings under shadows which result in a high number of false negatives.

Despite these specific challenges and rare instance segmentation issues, the overall performance of the model when evaluated on the validation set of 4075 images is highly favourable. The high values of the evaluation metrics, especially IoU, also indicate that the proposed model can segment buildings well within the feature edges and there is no region loss except for when the building itself is under a shadow. When compared with other deep learning-based approaches, the proposed model increases the average IoU to 0.80 and average F1 score to 0.86. Table 3 shows the overall evaluation metrics of the model for the validation set as well as a comparison of those metrics with other studies on the same IAIL dataset.

**Fig. 7** City-wise prediction metrics from the IAIL dataset validation part

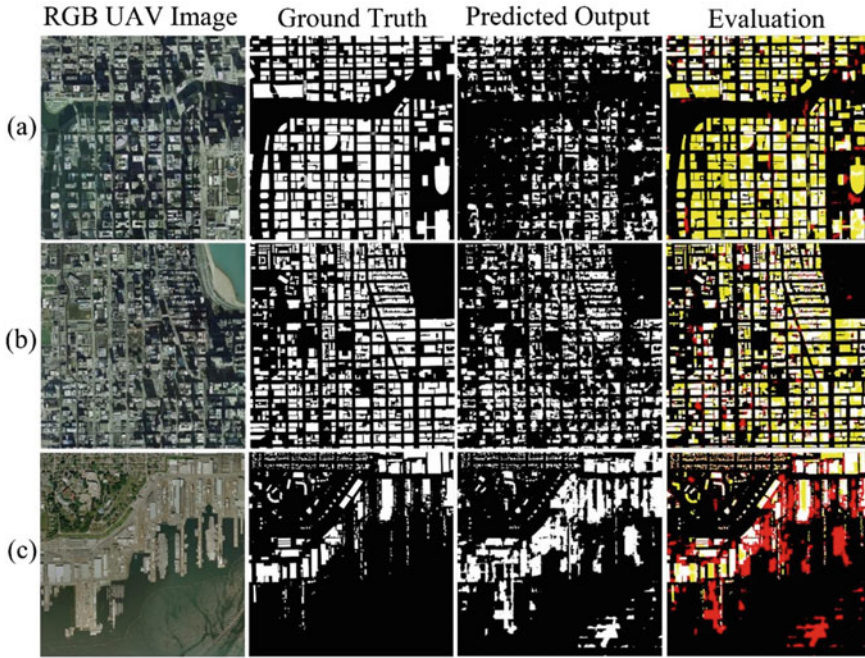


Fig. 8 Select instances where buildings are covered under shadows, leading to high false negative rate. First column is RGB input to the model, second column is model prediction for building segmentation, third column is ground truth and fourth column is evaluation image showing TP (white), TN (black), FP (red) and FN (yellow). **a, b** From Chicago, USA, **c** from Kitsap County

Table 3 Overall metrics of the proposed approach and their comparison with existing approaches

Method	Proposed Res-U-Net	Dilated spatial pyramid pooling [25]	GAN-SCA [27]	MSMT-Stage-1 [26]	AMLL [28]	Dilated CNN [29]
	0.965	0.894	0.966	0.961	0.959	0.928
	0.883	–	–	–	–	–
	0.861	–	–	–	–	–
	0.88	–	–	–	–	0.83
	0.193	–	–	–	–	–
	0.230	–	–	–	–	–
	93.42	–	–	–	–	–
	0.80	–	0.777	0.733	0.725	0.710

5 Conclusion

In this research work, building extraction from UAV imagery was explored using deep learning and transfer learning methodology. A Res-U-Net architecture consisting of U-Net blocks initialized with pre-trained ResNet34 weights and was used to learn building extraction from the IAIL dataset. The combination of ResNet and U-Net was used in an attempt to overcome the problems of blurred building boundaries and limited spectral resolution in building extraction. Moreover, a combined loss function that accounts both for the building region, as well as building boundaries, was used to train the proposed Res-U-Net. The model was trained and validated on 180 images from across five different cities of US and Austria. These images depicted high variance in terms of urban density and dominant land cover of the image. The proposed model was successfully able to segment buildings in all cases with rare instance segmentation issues. Model performance was measured using quantitative metrics of confusion matrix as well as object-based metrics such as branching factor, miss factor and IoU. When comparing these metrics with those of existing deep learning-based methods, highly favorable results were noted. Specific challenges such as extracting buildings lying under shadow and excluding turbid/active waterbody as a building were also identified and are open for research.

References

1. Momm H, Easso G (2011) Feature extraction from high-resolution remotely sensed imagery using evolutionary computation. *Evol Algorithms*. <https://doi.org/10.5772/15915>
2. Gomez-Chova L, Camps-Valls G, Munoz-Mari J, Calpe J (2008) Semi-supervised image classification with huberized Laplacian support vector machines. *IEEE Geosci Remote Sens Lett* 5(3):336–340. <https://doi.org/10.1109/ICET.2013.6743545>
3. Benediktsson JA, Pesaresi M, Arnason K (2003) Classification and feature extraction for remote sensing images from urban areas based on morphological transformations. *IEEE Trans Geosci Remote Sens* 41(9) PART I:1940–1949. <https://doi.org/10.1109/TGRS.2003.814625>
4. Sowmya A, Trinder J (2000) Modelling and representation issues in automated feature extraction from aerial and satellite images. *ISPRS J Photogramm Remote Sens* 55(1):34–47. [https://doi.org/10.1016/S0924-2716\(99\)00040-4](https://doi.org/10.1016/S0924-2716(99)00040-4)
5. Zou Q, Ni L, Zhang T, Wang Q (2015) Deep learning based feature selection for remote sensing scene classification. *IEEE Geosci Remote Sens Lett* 12(11):2321–2325. <https://doi.org/10.1109/LGRS.2015.2475299>
6. Huang Z, Cheng G, Wang H, Li H, Shi L, Pan C (2016) Building extraction from multisource remote sensing images via deep deconvolution neural networks. Huang Z, Cheng G, Wang H, Li H, Shi L, Pan C (2016) National laboratory of pattern recognition (NLPR) Institute of Au. In: 2016 IEEE International geoscience remote sensing symposium, pp 1835–1838
7. Karathanassi V, Iossifidis C, Rokos D (2000) A texture-based classification method for classifying built areas according to their density. *Int J Remote Sens* 21(9):1807–1823. <https://doi.org/10.1080/014311600209751>
8. Eskandarpour R, Khodaei A (2018) Leveraging accuracy-uncertainty tradeoff in SVM to achieve highly accurate outage predictions. *IEEE Trans Power Syst* 33(1):1139–1141. <https://doi.org/10.1109/TPWRS.2017.2759061>

9. Li W, He C, Fang J, Fu H (2018) Semantic segmentation based building extraction method using multisource GIS map datasets and satellite imagery. In: IEEE Computer society conference on computer vision and pattern recognition workshops, December 2018, vol 2018-June, pp 233–236. <https://doi.org/10.1109/CVPRW.2018.00043>
10. Cote M, Saeedi P (2013) Automatic rooftop extraction in nadir aerial imagery of suburban regions using corners and variational level set evolution. *IEEE Trans Geosci Remote Sens* 51(1):313–328. <https://doi.org/10.1109/TGRS.2012.2200689>
11. Huang X, Zhang L (2011) A multidirectional and multiscale morphological index for automatic building extraction from multispectral geospatial imagery. *Photogramm Eng Remote Sens* 77(7):721–732. <https://doi.org/10.14358/PERS.77.7.721>
12. Huang X, Zhang L (2012) Morphological building/shadow index for building extraction from high-resolution imagery over urban areas. *IEEE J Sel Top Appl Earth Obs Remote Sens* 5(1):161–172. <https://doi.org/10.1109/JSTARS.2011.2168195>
13. Zhu XX et al (2017) Deep learning in remote sensing: a review. <https://doi.org/10.1109/MGRS.2017.2762307>
14. Zhang L, Xia GS, Wu T, Lin L, Tai XC (2016) Deep learning for remote sensing image understanding. *J Sensors* 2016. <https://doi.org/10.1155/2016/7954154>
15. Erdem F, Avdan U (2020) Comparison of different U-net models for building extraction from high-resolution aerial imagery. *Int J Environ Geoinformatics* 7(3):221–227. <https://doi.org/10.30897/ijegeo.684951>
16. Chollet F (2018) Deep learning mit python und keras: das praxis-handbuch vom Entwickler der KerasBibliothek. MITP-Verlags GmbH & Co, KG
17. Li L, Liang J, Weng M, Zhu H (2018) A multiple-feature reuse network to extract buildings from remote sensing imagery. *Remote Sens* 10:1350, 10(9):1350. <https://doi.org/10.3390/RS10091350>
18. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. <http://www.robots.ox.ac.uk/>. Accessed 11 Jan 2021
19. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, December 2016, vol 2016–December, pp 770–778. <https://doi.org/10.1109/CVPR.2016.90>
20. Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H (2018) Encoder-decoder with Atrous separable convolution for semantic image segmentation. In: Lecture notes in computer science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), February 2018, vol 11211 LNCS, pp 833–851. https://doi.org/10.1007/978-3-030-01234-2_49
21. Yang H, Wu P, Yao X, Wu Y, Wang B, Xu Y (2018) Building extraction in very high resolution imagery by dense-attention networks. *Remote Sens* 10(11):1768. <https://doi.org/10.3390/rs10111768>
22. Badrinarayanan V, Kendall A, Cipolla R (2017) SegNet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans Pattern Anal Mach Intell* 39(12):2481–2495. <https://doi.org/10.1109/TPAMI.2016.2644615>
23. Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. LNCS (Including Subseries Lect Notes Artif Intell Lect Notes Bioinformatics) 9351:234–241. https://doi.org/10.1007/978-3-319-24574-4_28
24. Abdollahi A, Pradhan B, Alamri AM (2020) An ensemble architecture of deep convolutional Segnet and Unet networks for building semantic segmentation from high-resolution aerial images. *Geocarto Int*. <https://doi.org/10.1080/10106049.2020.1856199>
25. Morocho-cayamcela ME (2020) Increasing the segmentation accuracy of aerial images with dilated spatial pyramid pooling. *ELCVIA Electron Lett Comput Vis Image Anal* 19(2):17–21
26. Marcu A, Costea D, Slusanschi E, Leordeanu M (2018) A multi-stage multi-task neural network for aerial scene interpretation and geolocalization
27. Pan X et al (2019) Building extraction from high-resolution aerial imagery using a generative adversarial network with spatial and channel attention mechanisms. *Remote Sens* 11(8):917. <https://doi.org/10.3390/rs11080917>

28. Huang B et al (2018) Large-scale semantic classification: outcome of the first year of inria aerial image labeling benchmark. In: International Geoscience Remote Sensing Symposium, vol 2018 July, pp 6947–6950. <https://doi.org/10.1109/IGARSS.2018.8518525>
29. Khoshboresh-Masouleh M, Alidoost F, Arefi H (2020) Multiscale building segmentation based on deep learning for remote sensing RGB images from different sensors. *J Appl Remote Sens* 14(03):1. <https://doi.org/10.1117/1.jrs.14.034503>
30. Maggiori E, Tarabalka Y, Charpiat G, Alliez P (2017) High-resolution aerial image labeling with convolutional neural networks. *IEEE Trans Geosci Remote Sens* 55(12):7092–7103. <https://doi.org/10.1109/TGRS.2017.2740362>
31. Iglovikov V, Shvets A (2021) TeraNet: U-Net with VGG11 encoder pre-trained on imagenet for image segmentation. <http://arxiv.org/abs/1801.05746>. Accessed 01 Mar 2021
32. Lamba H (2019) Understanding semantic segmentation with UNET. *Towards Data Science*. <https://towardsdatascience.com/understanding-semantic-segmentation-with-unet-6be4f42d4b47>. Accessed 21 Jan 2021
33. Zhang Z, Sabuncu MR (2018) Generalized cross entropy loss for training deep neural networks with noisy labels
34. Sudre CH, Li W, Vercauteren T, Ourselin S, Cardoso MJ (2017) Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. *Lect Notes Comput Sci (Including Subser Lect Notes Artif Intell Lect Notes Bioinformatics)*, vol 10553 LNCS, pp 240–248. https://doi.org/10.1007/978-3-319-67558-9_28