Deep Learning with Semantic Segmentation Approach for Building Rooftop Mapping in Urban Irregular Housing Complexes

Edy Irwansyah
Computer Science Department, School of Computer Science, Bina Nusantara University, Indonesia
edirwan@binus.ac.id (corresponding author)

Alexander A.S. Gunawan
Computer Science Department, School of Computer Science, Bina Nusantara University, Indonesia
aagung@binus.edu

Hady Pranoto
Computer Science Department, School of Computer Science, Bina Nusantara University, Indonesia
hadypranoto@binus.ac.id

Fabian Surya Pramudya
Mathematic Department, School of Computer Science, Bina Nusantara University, Indonesia
fabian.pramudya@binus.edu

Lucky Fakhriadi
PT Deira Sygisindo, Jakarta Selatan, Indonesia
luckyf@paruhanggang.com

Received: 20 November 2024 | Revised: 26 December 2024 | Accepted: 29 December 2024

 DOI: https://doi.org/10.48084/etasr.9670

ABSTRACT

This research investigates the application of the Deep Learning (DL) U-Net architecture for building rooftop segmentation in densely populated urban areas with irregular housing patterns. The research explores the effectiveness of two loss functions - Binary Cross Entropy (BCE) and Dice Loss (DLs) - to optimize the segmentation accuracy. The present study utilized Small-Format Aerial Photography (SFAP) images processed into orthophotos with a final ground sampling distance of 5 cm. The study area, located in Bogor, Indonesia, features both regular and irregular housing patterns, making it an ideal testing ground for the segmentation model. The U-Net model, having been utilized EfficientNetB6 as the encoder and having been trained with augmented data, demonstrated stable performance across metrics, such as accuracy, precision, recall, and F1-score. The results show that the DLs function outperformed BCE, achieving an average Intersection over Union (IoU) score of 96.8% compared to the 87% score for BCE, indicating that DLs is more effective for this application. The study further enhances the segmentation results by converting the raster data into a vector format using the Ramer-Douglas-Peucker (RDP) algorithm, which simplifies and smooths the polygonal shapes of the segmented rooftops. The combination of the U-Net, DLs and RDP algorithm provides high accuracy results and high usability of the segmentation outputs in practical applications, such as urban planning and disaster management scenarios where accurate rooftop delineation is critical.

Keywords-deep learning; building rooftop; semantic segmentation; drone data; urban mapping

I. INTRODUCTION

DL has brought transformative advantages to building rooftop mapping, enabling the rapid creation of high-definition maps through precise rooftop delineation at the city scale using high-resolution satellite and aerial images [1, 2]. These methods have proven both efficient and accurate at extracting building rooftops, even when the training data are limited [2, 3]. DL architectures, such as U-Net and Fully Convolutional Networks (FCNs), have exhibited a notable ability to converge

quickly and achieve high accuracy when applied to Very High Spatial Resolution (VHSR) images, making them ideal for this application [4]. The applications of DL in rooftop mapping have been utilized to support urban applications, disaster management, and sustainable urban planning [3, 5, 6]. DL has been integrated with Geographic Information Systems (GIS) to accurately evaluate the rooftop solar potential in urban areas [7].

Despite these advancements, challenges remain in optimizing DL for rooftop mapping. A primary issue is scale variance, which limits the delineation accuracy, particularly for smaller buildings [1]. Furthermore, the automatic labeling of the training data and the generalizability of models across different environments remain unresolved challenges [2]. Additionally, the effect of the data volume on model performance is an area that warrants further investigation [4]. Another problem in rooftop mapping with imagery data is how to produce output data in the form of vectors from the segmentation results with regular and smooth shapes, given the nature of the raster-based input data. One solution is to implement a simple polygonized algorithm. The polygonized algorithm stands out for its simplicity, speed, adaptability, and efficiency, making it a robust choice for various applications in computational geometry and surface modeling [8-10].

This research aims to implement the U-Net algorithm with two different loss functions, namely, BCE and DLs to produce building roof segmentation in irregular housing complexes. It also employs the RDP polygonized algorithm to produce smooth vector output data for use in GIS.

II. RELATED WORK

A. Deep Learning for Building Rooftop Mapping

DL has potential applications in building roof mapping, but several challenges have hampered its effectiveness. These challenges include shadows and tree-covered buildings [11, 12], and irregular or complex building shapes [13]. Traditional model performance often struggles with irregular shapes during vectorization, leading to inaccuracies in the final result [11]. Complex buildings shading and various roof obstructions can lead to a suboptimal performance [14]. Incorporating prior knowledge of the building functions significantly improves detection accuracy, and utilizing fast learning to incorporate the building types into the model improves the identification of rooftops suitable for applications, such as photovoltaic energy development [15]. Another challenge in the use of DL is the issue of computational efficiency. The high computational demands of the DL models can be prohibitive. Low complexity models, such as the U-Net variants, have been proposed to reduce the training time while maintaining accuracy [12]. Authors in [16] proposed refining the U-Net elements and using ensemble models to improve the segmentation accuracy, whereas authors in [17] proposed addressing this problem by using city-specific models, optimal band selection, and automatic thresholding for segmentation. As mentioned in [18], the availability of publicly accessible high-resolution images, data labels, and training datasets also becomes another challenge. This limitation can cause high costs in data provisioning and in the labor-intensive manual labeling process [19], whereas addressing data sparsity and quality requires enhancing the training datasets [20].

Several strategies can be employed to enhance the accuracy of the DL-based rooftop mapping, focusing on model architecture optimization, data augmentation, and improved post-processing techniques. These approaches address the inherent challenges of complex urban environments and diverse rooftop geometries, with model architecture optimization being performed utilizing advanced neural networks [19] and hybrid models [12]. The use of models, such as Asymmetric Neural Network (ANN) and Dual Attention Network (DANet), has shown significant improvements in accuracy, with ANN achieving 96% in rooftop detection [19]. The integration of low complexity models, such as UNet-AstPPD, which incorporates Atrous Spatial Pyramidal Pooling (ASPP), can improve the feature selection while maintaining efficiency [12]. Dataset enrichment, performed in [21], uses data augmentation techniques to create a more robust training dataset, which is critical for models, such as Mask Region-based Convolutional Neural Network (Mask R-CNN), to better generalize under varying conditions [21]. The implementation of holistic edge classification and planar graph reconstruction can refine the delineation of complex roof boundaries and address issues of irregular polygon shapes in post-processing [11]. Table I displayes the most important research using DL and satellite / SFAP imagery data for building rooftop segmentation.

B. Ramer-Douglas-Peucker Algorithm

The RDP algorithm, also known as the Douglas-Peucker algorithm, is widely employed for polyline simplification in vector graphics and map generalization. The RDP algorithm effectively reduces the number of vertices in a polyline, which helps to compress the data without significant loss of detail, and improved versions of the RDP algorithm, such as those that incorporate radial distance constraints, have demonstrated better time efficiency compared to the original algorithm [22]. Parallel implementations of the RDP algorithm on multi-core processors significantly improve performance, enabling real-time simplification and display of the vector data [23, 24]. Enhanced versions of the RDP algorithm, like the Gestalt-based Douglas-Peucker (GDP) algorithm, maintain higher data quality and ensure that the area of simplified polygons remains unchanged [25]. Modifications to the RDP algorithm can preserve the topological structure and shape characteristics of contours, preventing problems, such as intersection or self-intersection [26]. In summary, the RDP algorithm's main advantages, shown in Table II, are its ability to efficiently compress data, improve processing time, support real-time applications, maintain data quality, and preserve topological integrity. These advantages make it a valuable tool in various applications, particularly in GIS and vector graphics simplification.

III. METHODOLOGY

The research consists of two main stages, defined as the building roof segmentation and polygonization of the raster segmentation results into vector polygons. The segmentation stage consists of the sub-stages of input data and pre-

processing, training data with the U-Net algorithm using two different loss functions, the testing data stage, and finally the data validation stage. The next stage is polygonization using the RDP algorithm to obtain the result in the form of smoothed vector data, which be can then used in various GIS tools. Figure 1 depicts the research workflow.

TABLE I. PREVIOUS RESEARCH ON THE IMPLEMENTATION OF DL IN ROOF MAPPING USING SATTELITE OR AERIAL/UAV DATA

Reference	Research gap	DL model	Dataset	Result
[11]	Delineating complex and irregular polygon shapes	Attention-based neural network; Planar graph reconstruction for building roof plane extraction	VHR aerial images, building footprints, the building internal roof planes and LIDAR point cloud	Combined dataset model achieved F-score of 0.43 / Individual datasets had lower F-scores: 0.37 and 0.32
[12]	The diversity and complexity of building structures	UNet – AstPPD; UNetVasyPPD with VGG backbone	Satellite imagery	UNet-AstPPD shows better accuracy and DLs / Training times: 25.44 and 29.23 minutes
[13]	Effectiveness of DL models for building extraction is unexplored / Critical features for distinguishing buildings from other land cover types are unclear	Convolutional Neural Network (CNN)	High-resolution RGB images / Geographic Object-Based Image Analysis (GEOBIA) framework	Some shallow learning classifiers perform similarly to DL models / Critical features for building extraction are identified
[19]	-	ANN; DANet; PP-LiteSeg; Deeplab3	Large dataset of labeled UAV rooftop building images / Superpixel regions assigned binary labels for rooftop presence	ANN model achieved the highest accuracy with 96% / DANet model followed with an accuracy of 95.09%
[15]	Most methods ignore building type prior information / Deeplabv3+ backbone presents suboptimal performance without prior knowledge	Pre-trained semantic segmentation network; Deeplabv3+, detects rooftops / Prompt learning incorporates building types as prior knowledge	-	Achieved 81.18% accuracy in rooftop PV potential prediction / Improved IoU by 10.97% with prior knowledge
[21]	The paper does not specify metrics indicating improved accuracy of DL techniques	Mask R-CNN / Data augmentation techniques for model training	Annotated satellite images of solar PV rooftops created / Augmented dataset enhances generalization and prevents overfitting	Superior performance in detecting solar PV installations / Exceptional capability in delineating individual solar panels

TABLE II. SOME ADVANTAGES AND DESCRIPTIONS OF RDP ALGORITHMS

Advantage	Description	References
Compression efficiency	Reduces vertices while maintaining shape	[22]
Time efficiency	Improved algorithms offer better time efficiency	[22]
Real-time performance	Parallel implementations enhance real-time processing	[23, 24]
Data quality	Enhanced algorithms maintain higher data quality and polygon area consistency	[25]
Topological integrity	Prevents intersection or self-intersection in contours	[26]

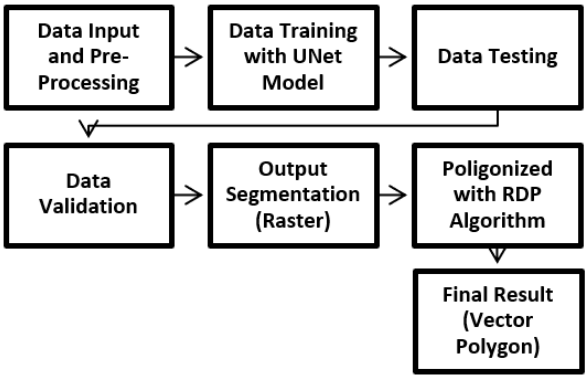


Fig. 1. Research workflow.

A. Dataset and Research Area

The primary dataset consists of SFAP images processed into orthophotos with a final ground sampling distance of 5 cm. Raw images are acquired using a CHCNAV P330 PRO VTOL unmanned aerial vehicle equipped with a Sony A7RIV full-frame camera sensor, with 80% front overlap and 60% side overlap for image acquisition. The flight altitude for aerial imagery acquisition is averaged at 400 meters, with the flight path adjusted to the direction of the nearest local air force base (Atang Sanjaya airport, Bogor, Indonesia). Orthophoto processing was done using the Agisoft software with a final ground sampling distance of 5 cm.

The location of the research area is in a densely populated residential area with irregular building patterns in the city of Bogor, West Java, Indonesia (6°32'22"S, 106°48'40"E, and 6°32'53"S 106°47'57"E). Figure 2 illustrates two different building characteristics, the first characteristic (a) is a regular residential area, but the buildings are still attached to each other, besides the roof pattern has the same color; whereas the second characteristic (b) is an irregular densely populated residential area, which has irregular house and roof positions.

B. Data Pre-processing

Before the image data are labeled, the orthophotos are pre-processed in the QGIS software to subdivide the images into smaller chunks of 1024 × 1024 pixels. Each scene produces approximately 150 images in ".tiff" format. The images are labeled into two main criteria, building and non-building, as

portrayed in Figure 3(b) in white and black color, respectively. The feature extraction process will rely on the accuracy of this labeling process, especially since the size of the chunks is rather small.

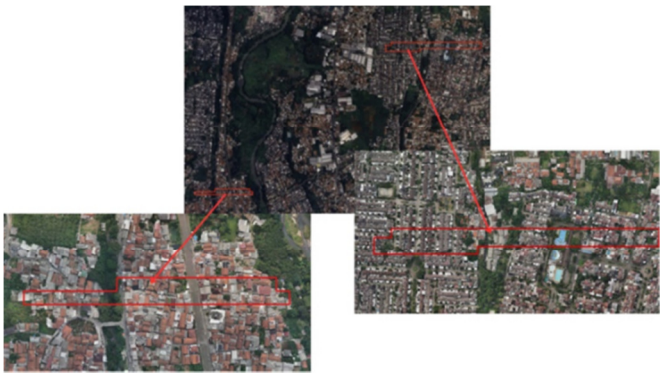
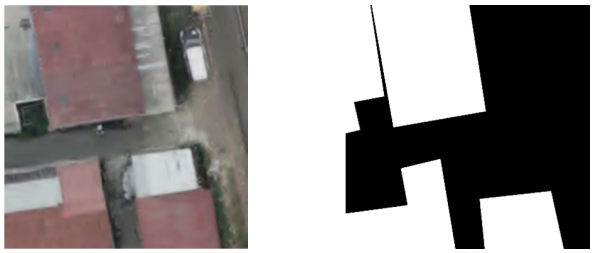


Fig. 2. Visualization of drone data on the research area in Bogor City, Indonesia.



(a) (b)

Fig. 3. Drone imagery labeling data: (a) Example of drone data with a building rooftop view and (b) the result of imagery labeling process.

C. Model Development

The U-Net model uses EfficientNetB6 as the encoder, as can be seen in Figure 4, with a total of 22.7 million parameters, and is trained utilizing a 6GB RTX 3060 GPU and 16GB memory. The U-Net model receives 512 × 512-pixel input, by first resizing the training image. Augmentations, such as vertical flip and horizontal flip, are performed, resulting in a 300-image dataset, which is 90% for training (270 images) and 10% for validation (30 images). The deployed model has some adjustments from the original U-Net model, which in this study were carried out using 32, 64, 128, 256, and 512 kernels, with the aim of reducing computation during training, due to limited hardware resources.

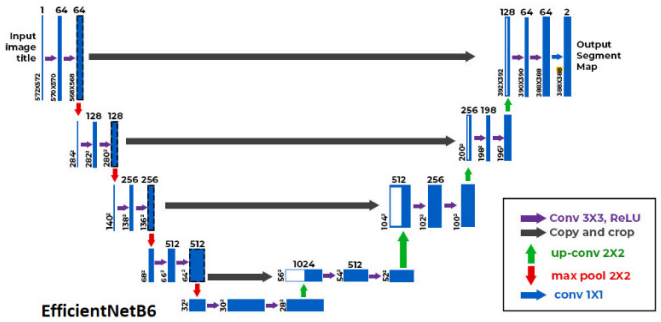


Fig. 4. U-Net model architecture with EfficientNetB6.

D. Polygonized Raster to Vector

The technique implemented in this research to convert the segmentation results, which were in the form of raster data, into the vector polygon form uses the RDP algorithm [27] as a simplified polygon by reducing the points on the polygon line. Simplification is done so that the polygon results have firmer lines, since polygons are generated from segmentation contours that have a lot of curvature, which tends to be non-smooth. The pseudocode for the RDP algorithm is:

```
function DouglasPeucker(PointList[],
epsilon)
# Find the point with the maximum distance
Dmax = 0
index = 0
end = length(PointList)
for i = 2 to (end - 1) {
   d = perpendicularDistance(PointList[i],
   Line(PointList[1], PointList[end])
   if (d > dmax) {
      index = i
      dmax = d
   }
}
ResultList[] = empty;
# If max distance is greater than epsilon,
recursively simplify
if (dmax > epsilon) {
   # Recursive call
   recResults1[] =
   DouglasPeucker(PointList[1...index],
   epsilon)
   recResults2[] =
   DouglasPeucker(PointList[index...end],
   epsilon)
   # Build the result list
   ResultList[] =
   {recResults1[1...length(recResults1) -
   1],
   recResults2[1...length(recResults2)]}
} else {
   ResultList[] = {PointList[1],
   PointList[end]}
}
# Return the result
return ResultList[]
```

E. Training Metrics and Loss Function

The research experiment compared two loss functions, BCE and DLs, to see which one has better prediction results by evaluating the IoU value. These loss functions have been widely used as the standard procedure to improve the segmentation result in binary classification tasks [28]. The formula of each loss function is:

$$BCE = -\frac{1}{N}\sum_{i=1}^{N}\left[y_{true} * \log\left(y_{pred}\right) + \left(1 - y_{true}\right) * \log\left(1 - y_{pred}\right)\right] \quad (1)$$

$$Dice\ Loss = 1 - \frac{2|A \cap B|}{|A| + |B|} \quad (2)$$

Additional metrics used for the evaluation are the accuracy, precision, recall, and F1-score with the following formulas:

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \quad (3)$$

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

$$F1\text{-}score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (6)$$

where TP is true positive, TN is true negative, FP is false positive, and FN is false negative.

IV. RESULTS

A. Training Model Results

During model training, this study uses the checkpoints to save the best model based on the loss value with additional learning rate callbacks being applied, aiming to reduce the learning rate value when the improvements are not significant. The minimum learning rate value is set to 1e-8. The basic U-Net model is trained with a maximum number of 30 epochs, batch size 1, learning rate 1e-4, and Adam optimizer. EfficientNetB6 is selected as the encoder, and DLs and BCE are used as the loss function.

The accuracy training value shows that BCE has a slightly higher value than DLs, as evidenced in Figure 5. At epoch 8, DLs accuracy validation experienced the lowest decline at 0.89 and began to stabilize at epochs 17 to 30 with an average value of 0.93. As for the accuracy, BCE exhibits a stable value at epochs 7 to 30 with an average value of 0.93. The average training time for the models that use BCE is 36 minutes, while the models that utilize DLs take 35 minutes, which demonstrates the relatively similar use of computational power. Based on the training loss, the value of the BCE loss validation is higher, with its lowest point occurring at epoch 9 with a value of 0.17, while the highest value occurrs at epoch 12 with a value of 0.307. In contrast, the DLs validation exhibits a low value, with the lowest value being observed at epoch 10 and the highest, 0.099, at epochs 7 to 8, as depicted in Figure 6. The results of the training precision value reveal that the BCE precision validation has a slightly higher value, with the highest, 0.968, being at the epoch 18 and the lowest, 0.943, being at epoch 7. DLs validation precision has the lowest value, 0.91, at epoch 7 and the highest, 0.96, at epoch 13, as illustrated in Figure 7. Regarding the recall of the U-Net model training results, the DLs recall validation has the lowest value, 0.84, at epoch 8 and the highest, 0.93, at epoch 10 whereas the BCE recall validation has the higher lowest point at epoch 4 with a value of 0.86, and the highest at epoch 22 with a value of 0.91, as depicted in Figure 8.

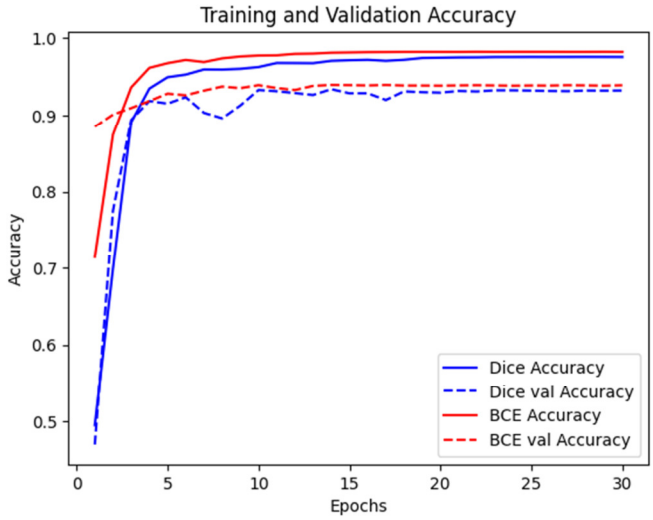


Fig. 5. Training and validation accuracy with BCE and DLs.

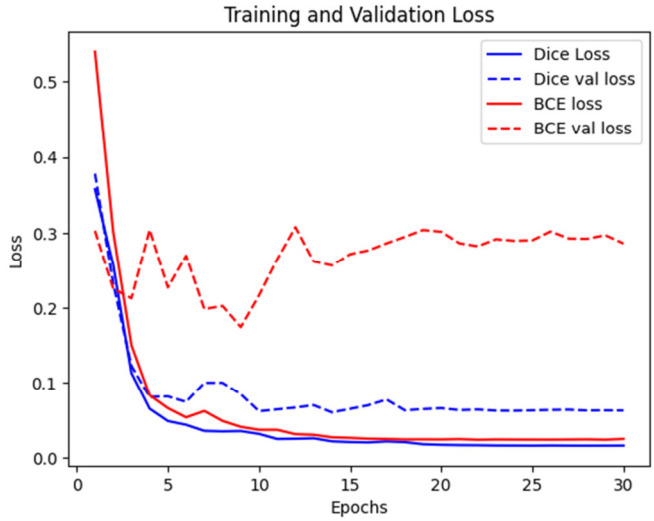


Fig. 6. Training and validation loss with BCE and DLs.

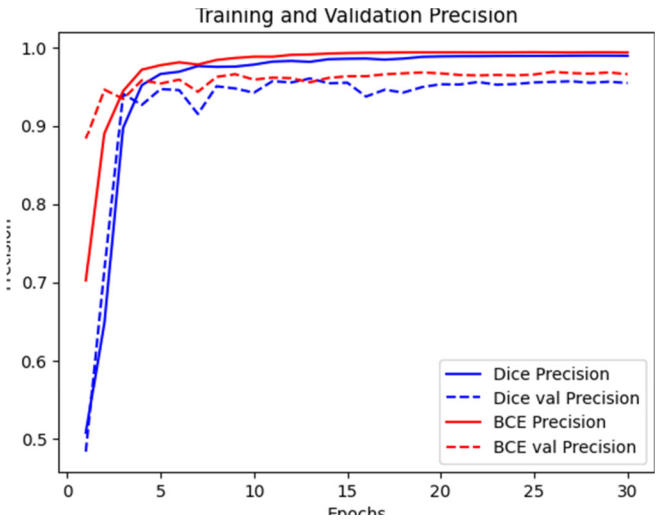


Fig. 7. Training and validation precision with BCE and DLs.

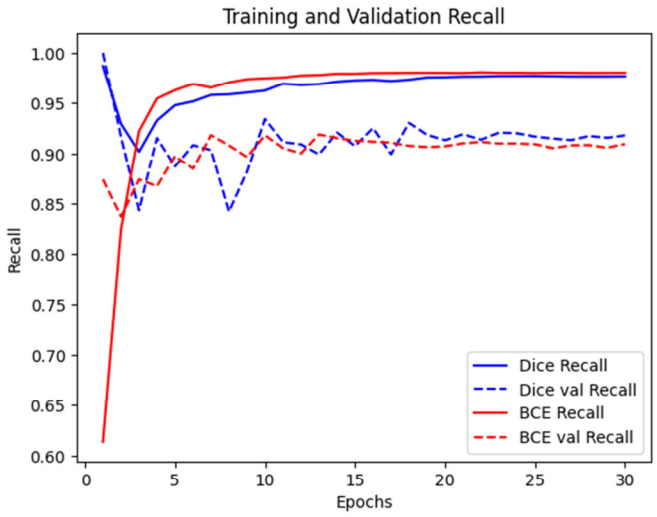


Fig. 8. Training and validation recall with BCE and DLs.

The F1-score analysis shows that DLs F1-score validation is the lowest at epoch 8 with a value of 0.89 and the highest at epoch 10 with a value of 0.938, whereas BCE F1-score validation is the lowest value at epoch 6 with a value of 0.92 and the highest at epoch 10 with a value of 0.93, as can be seen in Figure 9.

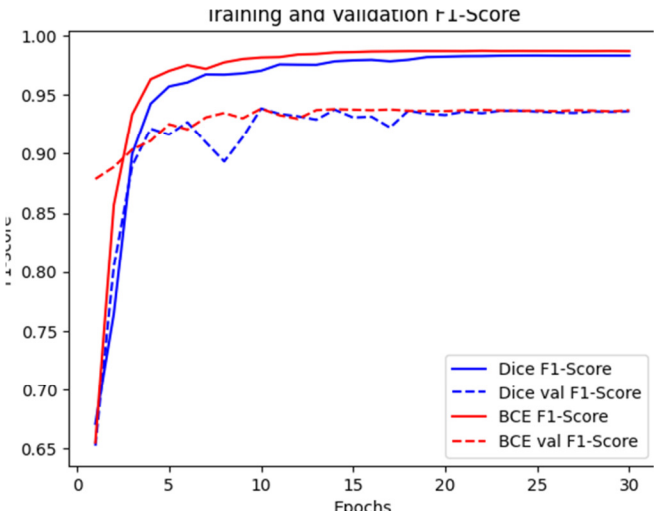


Fig. 9. Training and validation F1-score with BCE and DLs.

In the training part, the accuracy, loss, precision, recall, and F1-score showed good and stable results during the training process. At epoch 10, the F1-score validation value of DLs shows a momentary decrease, with a value of 0.893, and the validation F1-score value of BCE has a stable condition during training.

B. Evaluation Model Results

In the model evaluation, the IoU value was used as the main parameter to evaluate the segmentation results of the U-Net model, by comparing BCE and DLs, as demonstrated in Table III. A threshold of 0.75 was utilized to perform the segmentation in both models.

TABLE III. BUILDING ROOF SEGMENTATION RESULTS USING U-NET WITH BCE AND DLS

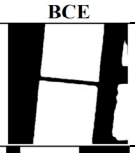
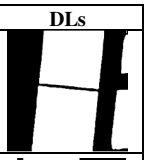
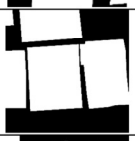
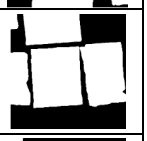
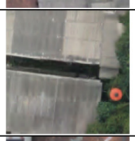
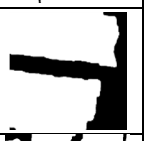
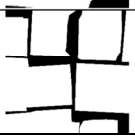
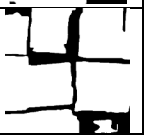
No	RGB Image	Label Image	BCE	DLs
1				
2				
3				
4				
5				

Table IV shows the segmentation results of the models trained on 50 test images with two different loss functions. From the calculation of the IoU values for both models, the U-Net with DLs as a loss function produces segmentations with higher IoU values compared to BCE, with an average value of 87% for the model wih BCE and 96.8% for the model with DLs.

TABLE IV. IOU SCORE COMPARISON BETWEEN BCE AND DLS BASED ON TABLE III

Image no.	BCE IoU (%)	DLs IoU (%)
1	94.09	98.02
2	71.77	96.12
3	92.51	96.58
4	90.72	94.26
5	85.80	93.85
Average	87	96.8

After segmenting the test image, the model segmentation results are converted into a polygon form by applying the RDP algorithm to perform vector smoothing. The results of vector polygon without simplification appear more indented than those with simplification, where the polygon line shape is straighter and smoother, as illustrated in Figure 10. The use of RDP allows for clearer boundaries between the segments of irregular buildings and buildings with a diagonal edge relative to the image axis. While the results are not noticeable for most regular buildings, this study's results exhibit that the combination of these methods is highly recommended to improve the segmentation results in more complex cases.

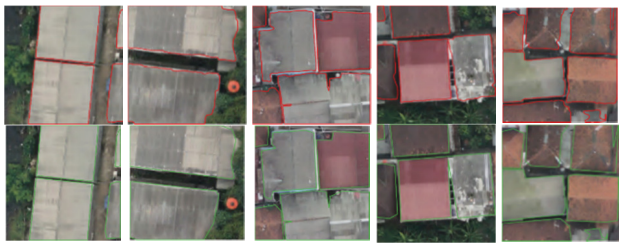


Fig. 10. Comparison of polygonized results between U-Net(EfficientNetB6)+DLs (a) and U-Net(EfficientNetB6)+DLs +RDP (b).

V. CONCLUSION

The Deep Learning (DL) segmentation method using U-Net with EfficientNetB6 as the encoder has demonstrated improved results for building rooftop mapping, outperforming previous models, such as the Dual Attention Network (DANet) and Asymmetric Neural Network (ANN). In experiments comparing two loss functions, Dice Loss (DLs) achieved a significantly higher average Intersection over Union (IoU) score of 96.8% compared to the 87% obtained with Binary Cross Entropy (BCE), confirming the superior effectiveness of DLs for this application.

The integration of U-Net with the Ramer-Douglas-Peucker (RDP) algorithm effectively converts the raster-based segmentation results into refined vector formats, simplifying and enhancing the polygon shapes of segmented rooftops. This combination not only ensures high segmentation accuracy, but also enhances the practical usability of the output data for Geographic Information System (GIS) applications, making it a valuable tool in urban planning, infrastructure development, and other real-world contexts.

CONFLICT OF INTEREST

The authors declare that they have no known conflicts of interest that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY OF DATA AND MATERIAL

The dataset generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

REFERENCES

[1] H. He, L. Ma, and J. Li, "HigherNet-DST: Higher-Resolution Network With Dynamic Scale Training for Rooftop Delineation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–15, Feb. 2024, https://doi.org/10.1109/TGRS.2024.3362601.

[2] K. Sawa, I. Yalcin, and S. Kocaman, "Building Detection from SkySat Images with Transfer Learning: a Case Study over Ankara," *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, vol. 92, no. 2, pp. 163–175, Apr. 2024, https://doi.org/10.1007/s41064-024-00279-x.

[3] L. I. U. Wentao, L. I. Shihua, and Q. I. N. Yuchu, "Automatic Building Roof Extraction with Fully Convolutional Neural Network," *Journal of Geo-information Science*, vol. 20, no. 11, pp. 1562–1570, Nov. 2018, https://doi.org/10.12082/dqxxkx.2018.180159.

[4] H. He *et al.*, "The Impact of Data Volume on Performance of Deep Learning Based Building Rooftop Extraction Using Very High Spatial Resolution Aerial Images," in *2021 IEEE International Geoscience and Remote Sensing Symposium*, Brussels, Belgium, 2021, pp. 1343–1346, https://doi.org/10.1109/IGARSS47720.2021.9553422.

[5] J. Yang, B. Matsushita, and H. Zhang, "Improving building rooftop segmentation accuracy through the optimization of UNet basic elements and image foreground-background balance," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 201, pp. 123–137, Jul. 2023, https://doi.org/10.1016/j.isprsjprs.2023.05.013.

[6] I. García-Aguilar, J. Galeano-Brajones, F. Luna-Valero, J. Carmona-Murillo, J. D. Fernández-Rodríguez, and R. M. Luque-Baena, "Prediction of Optimal Locations for 5G Base Stations in Urban Environments Using Neural Networks and Satellite Image Analysis," in *Bioinspired Systems for Translational Applications: From Robotics to Social Engineering: 10th International Work-Conference on the Interplay Between Natural and Artificial Computation,Proceedings, Part II*, Olhâo, Portugal, 2024, pp. 33–43, https://doi.org/10.1007/978-3-031-61137-7_4.

[7] G. Li *et al.*, "A district-scale spatial distribution evaluation method of rooftop solar energy potential based on deep learning," *Solar Energy*, vol. 268, Jan. 2024, Art. no. 112282, https://doi.org/10.1016/j.solener.2023.112282.

[8] M. Cermak and V. Skala, "Edge spinning algorithm for implicit surfaces," *Applied Numerical Mathematics*, vol. 49, no. 3–4, pp. 331–342, Jun. 2004, https://doi.org/10.1016/j.apnum.2003.12.011.

[9] M. Cermak and V. Skala, "Surface Curvature Estimation for Edge Spinning Algorithm," in *Computational Science - ICCS 2004: Proceedings of 4th International Conference on Computational Science, Part II*, Kraków, Poland, 2004, pp. 412–418, https://doi.org/10.1007/978-3-540-24687-9_52.

[10] D. Harbinson, R. Balsys, and K. Suffern, "Hybrid Polygon-Point Rendering of Singular and Non-Manifold Implicit Surfaces," in *2019 23rd International Conference in Information Visualization – Part II*, Adelaide, Australia, 2019, pp. 160–166, https://doi.org/10.1109/IV-2.2019.00039.

[11] C. Campoverde, M. Koeva, C. Persello, K. Maslov, W. Jiao, and D. Petrova-Antonova, "Automatic Building Roof Plane Extraction in Urban Environments for 3D City Modelling Using Remote Sensing Data," *Remote Sensing*, vol. 16, no. 8, Apr. 2024, Art. no. 1386, https://doi.org/10.3390/rs16081386.

[12] A. Ramalingam, V. Srivastava, S. V. George, S. Alagala, and M. L. Manickam, "Building rooftop extraction from aerial imagery using low complexity UNet variant models," *Journal of Spatial Science*, vol. 69, no. 3, pp. 773–800, Jul. 2024, https://doi.org/10.1080/14498596.2024.2302166.

[13] M. D. Hossain and D. Chen, "Performance Comparison of Deep Learning (DL)-Based Tabular Models for Building Mapping Using High-Resolution Red, Green, and Blue Imagery and the Geographic Object-Based Image Analysis Framework," *Remote Sensing*, vol. 16, no. 5, Mar. 2024, Art. no. 878, https://doi.org/10.3390/rs16050878.

[14] H. Ni *et al.*, "Enhancing rooftop solar energy potential evaluation in high-density cities: A Deep Learning and GIS based approach," *Energy and Buildings*, vol. 309, Apr. 2024, Art. no. 113743, https://doi.org/10.1016/j.enbuild.2023.113743.

[15] X. Han, J. Wang, X. Liu, J. Du, X. Bai, and R. Ji, "PromptNet: Prompt Learning for Roof Photovoltaic Potential Assessment," *Journal of Physics: Conference Series*, vol. 2755, no. 1, May 2024, Art. no. 012042, https://doi.org/10.1088/1742-6596/2755/1/012042.

[16] J. Yang, B. Matsushita, and H. Zhang, "Improving building rooftop segmentation accuracy through the optimization of UNet basic elements and image foreground-background balance," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 201, pp. 123–137, Jul. 2023, https://doi.org/10.1016/j.isprsjprs.2023.05.013.

[17] A. Churi and D. B. Megherbi, "Methods for Predictive Performance Improvement of Deep Learning Systems for Aerial Building Roof Detection with Multispectral Images," in *2023 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications*, Gammarth, Tunisia, 2023, pp. 1–6, https://doi.org/10.1109/CIVEMSA57781.2023.10231019.

[18] Z. Liu, H. Tang, L. Feng, and S. Lyu, "China Building Rooftop Area: the first multi-annual (2016–2021) and high-resolution (2.5 m) building rooftop area dataset in China derived with super-resolution segmentation

from Sentinel-2 imagery," *Earth System Science Data*, vol. 15, no. 8, pp. 3547–3572, Aug. 2023, https://doi.org/10.5194/essd-15-3547-2023.

[19] Z. K. Hussain, J. Congshir, Y. X. Xin, and M. R. e Mustafa, "A Comparative Study of PP-LiteSeg, Dual Attention Network, DeeplabV3p and Asymmetric Neural Network for Rooftop Detection in UAV Images." Preprints, Apr. 10, 2024, https://doi.org/10.20944/preprints202404.0705.v1.

[20] M. Buyukdemircioglu, R. Can, and S. Kocaman, "Deep Learning Based Roof Type Classification using Very High Resolution Aerial Imagery," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLIII-B3-2021, pp. 55–60, Jun. 2021, https://doi.org/10.5194/isprs-archives-XLIII-B3-2021-55-2021.

[21] P. Chaweewat, "Solar photovoltaic rooftop detection using satellite imagery and deep learning," in *2023 IEEE PES 15th Asia-Pacific Power and Energy Engineering Conference*, Chiang Mai, Thailand, 2023, pp. 1–5, https://doi.org/10.1109/APPEEC57400.2023.10561976.

[22] X. Wang, J. Zhang, and L. You, "A Douglas-Peucker Algorithm Combining Node Importance and Radial Distance Constraints," in *2021 3rd International Conference on Artificial Intelligence and Advanced Manufacture*, Manchester, United Kingdom, 2021, pp. 265–269, https://doi.org/10.1145/3495018.3495063.

[23] Jinsong M. A., Jie S., and Shoucheng X. U., "A Parallel Implementation of Douglas-Peucker Algorithm for Real-Time Map Generalization of Polyline Features on Multi-core Processor Computers," *Geomatics and Information Science of Wuhan University*, vol. 36, no. 12, pp. 1423–1426, Dec. 2011.

[24] J. Ma, S. Xu, Y. Pu, and G. Chen, "A real-time parallel implementation of Douglas-Peucker polyline simplification algorithm on shared memory multi-core processor computers," in *2010 International Conference on Computer Application and System Modeling*, Taiyuan, China, 2010, pp. V4-647-V4-652, https://doi.org/10.1109/ICCASM.2010.5620612.

[25] X. Song, C. Cheng, C. Zhou, and D. Zhu, "Gestalt-Based Douglas-Peucker Algorithm to Keep Shape Similarity and Area Consistency of Polygons," *Sensor Letters*, vol. 11, no. 6–7, pp. 1015–1021, Jun. 2013, https://doi.org/10.1166/sl.2013.2895.

[26] Z. Xie, H. Wang, and L. Wu, "The improved Douglas-Peucker algorithm based on the contour character," in *2011 19th International Conference on Geoinformatics*, Shanghai, China, 2011, pp. 1–5, https://doi.org/10.1109/GeoInformatics.2011.5981173.

[27] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *Cartographica*, vol. 10, no. 2, pp. 112–122, Dec. 1973, https://doi.org/10.3138/FM57-6770-U75U-7727.

[28] D. Patil and S. Jadhav, "Road Segmentation in High-Resolution Images Using Deep Residual Networks," *Engineering, Technology & Applied Science Research*, vol. 12, no. 6, pp. 9654–9660, Dec. 2022, https://doi.org/10.48084/etasr.5247.