ComTech: Computer, Mathematics and Engineering Applications, 16(2), December 2025, 127−138
DOI: 10.21512/comtech.v16i2.12089
P-ISSN: 2087-1244
E-ISSN: 2476-907X

Inception-ResNet-V2 The U-Net Encoder for Road Segmentation using Sentinel 2A

Bayu Yanuargi[1]*; Ema Utami[2]

[1-2]Program Doktoral, Fakultas Teknologi Informasi, Universitas Amikom,
Yogyakarta, Indonesia 60231
[1]bayu.yanuargi@students.amikom.ac.id; [2]ema.u@amikom.ac.id

Received: 29th August 2024/ **Revised:** 11th March 2025/ **Accepted:** 11th March 2025

How to Cite: Yanuargi, B., & Utami, E. (2025). Inception-ResNet-V2 the U-Net Encoder for road segmentation using Sentinel 2A. *ComTech: Computer, Mathematics and Engineering Applications, 16*(2), 127−138. https://doi.org/10.21512/comtech.v16i2.12089

Abstract **-** Updating road network maps is essential for transportation services, as incomplete or inaccurate maps can lead to inefficiencies and diminish service quality. The online transportation industry generates vast amounts of GPS data as drivers navigate, which is valuable for mapping road networks and improving traffic management. However, since drivers do not cover all roads, satellite imagery plays a crucial role in identifying areas that are not mapped. By combining GPS data as labels with satellite imagery, the extraction of new road networks becomes more accurate. This research employs a deep learning convolutional neural network with the U-Net architecture for road segmentation, allowing for the identification of new paths. Two different encoders are tested in this research: Inception-ResNet-V2 and a pure U-Net encoder. The Inception-ResNet-V2 encoder achieves an accuracy of 91.3%, while the pure U-Net encoder achieves 90.7%. In terms of Dice Loss, the models record values of 0.051 and 0.08, respectively. The research highlights the effectiveness of different U-Net encoders in road network segmentation. With high accuracy and low Dice Loss, this approach provides a reliable method for automatically updating road maps. It has potential applications in navigation systems, urban planning, and AI-driven intelligent transportation systems.

Keywords**:** convolutional neural network, U-Net, Inception-ResNet-V2, encoder, deep learning, computer vision, segmentation

I. INTRODUCTION

The update of the Road Network Map (RNM) is crucial for providing transportation services. However, if the map is incomplete or inaccurate, it can lead to poor functionality and a decline in service quality. For instance, incorrect distance calculations can lead to errors in price estimations and may also reveal hazardous travel routes. In the online transportation industry, such as Grab or Gojek, the complexity of the road network affects how quickly drivers reach their destinations. Consequently, drivers assigned solely based on the shortest straight-line distance may not be the most optimal choices. This is due to the inherent need for slight detours within the road network, which can result in extended travel times to pick-up locations (Liu et al., 2022).

In addition to serving the private sector, the road network plays a crucial role in government planning and policy development, as it is essential for the proper development of cities and regions. As primary transportation facilities, roads fulfill several essential functions. First, they facilitate the movement of people and goods. Second, they connect activity centers within and between cities (Luthfil et al., 2021).

To effectively support government and private planning as well as policy development, it is essential to frequently update the road network map, particularly in rapidly developing urban areas like South Jakarta, Indonesia. Additionally, the Indonesian Topographical Map, created by the Geospatial Information Agency (Badan Informasi Geospasial), serves as one of the primary tools used by the city government for spatial planning. This base map provides detailed land surface information developed in accordance with Indonesia's regulatory standards, ensuring the production of high-quality and integrated planning instruments (Pinuji et al., 2019).

To develop or update road network maps, two standard methods are used. The first method involves conducting field surveys, which utilize geodetic GPS to create accurate maps. According to Joubert et al. (2020), the accuracy levels for Low-Cost GNSS and GPS Geodetic measurements are between 10 and 21 cm and between 7 and 60 cm, respectively, with an

*Corresponding Author

average measurement time of 65 minutes per point for Low-Cost GNSS and 10 minutes for GPS Geodetic. This suggests that field surveys are expensive and have limited coverage, despite their high accuracy. The second method involves extracting satellite imagery, which is generally faster and more cost-effective than direct field surveys. However, this approach has its challenges, including difficulties in capturing a complete road network due to obstructions from trees and buildings, as well as potential confusion with other similar features, such as rice fields, rivers, and railroad tracks (Joubert et al., 2020).

To obtain accurate information, it is essential to support satellite imagery extraction with proper data labeling. The most common method for providing labeled data involves manual interpretation, where roads are drawn directly from the imagery. An alternative approach utilizes GPS tracking data to capture accurate road patterns. In a research conducted in Beijing, a Convolutional Neural Network (CNN) was employed to extract road data from satellite imagery. This process integrated both GPS data and satellite imagery, leading to several comparative conclusions. The results demonstrated very high precision for all variables involved, including satellite imagery, GPS data, and point data (Sun et al., 2019).

High-quality digital road maps are crucial for location-based services and smart-city applications. The vast and accessible GPS data generated by mobile devices is key to developing new mapping patterns. However, automated roadmap generation presents a challenge due to the low sampling rate and the issue of multi-level disparity, which means that the maps created have not yet met commercial standards (Chen et al., 2021).

Manually drawing over imagery is a standard practice used by some researchers. However, it demands significant time and resources. Deep learning technology presents new opportunities for extracting road networks from satellite imagery. Nonetheless, recent segmentation methods based on CNN face serious challenges regarding road connectivity. Road tracing techniques that rely on a single starting point often encounter problems, as certain areas may not be accurately identified (Wei et al., 2019).

Wei et al. (2019) conduct their research in two stages. The first step involves determining the starting point, which is divided into two parts: road segmentation and starting point detection. During this process, the Fully Convolutional Network (FCN) algorithm is utilized. The second stage focuses on searching for paths from multiple starting points using the CNN algorithm. According to the research's findings, starting points located at crossroads can lead to a slight increase in RoadTracer's performance (using a single starting point), with an improvement of approximately 0.2% in Intersection over Union (IoU). This suggests that a starting point that offers multiple potential directions can produce slightly better outcomes. In comparison, RoadTracer-M (utilizing multiple starting points) demonstrates significant advancements in road topology searches, achieving improvements of 10.4% and 8.7% in F1-Score and IoU, respectively. These improvements represent relative gains of 38.6% and 51.1% when compared to RoadTracer.

For geographic analysis, such as map inference, matching, and traffic detection, GPS data is essential. However, one of the main challenges is the limited availability of this data, particularly when privacy concerns arise. Additionally, the frequency of data collection can vary significantly, impacting both the quantity and quality of the information obtained. This inconsistency in data availability highlights that larger, higher-quality datasets are often concentrated in specific regions, particularly in developed countries such as China and the United States. These regions benefit from more advanced infrastructure, which enhances data collection and processing capabilities. Conversely, in other parts of the world, the availability of GPS data is often sparse or inconsistent, making it challenging to conduct detailed geographic analyses on a larger scale. The differences in data coverage across various areas highlight the need for improved global data accessibility and infrastructure to support more comprehensive geographic studies.

Grab position data is a leading source of information availability in Southeast Asia, particularly in Singapore and Indonesia. As of April 2019, these data sets boasted high accuracy with recording intervals of one second. In addition to their large volume, the GPS data provide valuable insights, including accuracy levels, bearing, and speed, which are essential for in-depth transportation analysis. They also include information such as the driver's unique ID and the types of vehicles and devices used (Android or iPhone) (Xu et al., 2020).

The use of location-based services that rely on GPS technology involves highly complex algorithms for GPS-based map creation, which can require substantial storage space. Research has been conducted on the application of the Othello Coordinated method, with the expectation that it can optimize the challenges associated with significant raster map storage and computational performance. In a recent experiment, the prediction accuracy reached an impressive 99.86% (J. Zhang et al., 2021).

This research aims to compare the accuracy of the U-Net architecture when using Inception-ResNet-V2 as the encoder versus when it is not. The U-Net architecture is inherently complex, and this modification aims to determine whether using an alternative encoder has a significant impact on the model's accuracy. Additionally, this research aims to demonstrate how GPS data can be utilized as labels to enhance the efficiency of the process, thereby eliminating the need for manual interpretation during the labeling phase.

II. METHODS

This research presents a six-step realignment method that can be repeated to enhance accuracy and precision. The process begins with trimming the GPS points and preparing the map for realignment. Next, the GPS points are aligned with public road segments. Once a GPS point is matched to a road segment, a new road segment is generated to best fit that point. Using these GPS-adjusted segments, the location of the new intersection is identified. Following this, the geometric characteristics of the original road segment and the rail characteristics obtained from the GPS are restored. This method aims to enhance the precision of mapping and can be iteratively applied to achieve better results. The overall research methodology is visually represented in Figure 1, which outlines the entire process in a step-by-step manner. This visual aid enhances the understanding of how each phase of the realignment process is interconnected, highlighting the sequential flow of the steps involved in the research.

In this research, two types of data are utilized: GPS data and Sentinel-2A satellite imagery. The GPS data, obtained from the ride-hailing driver app Grab in April 2019, consists of 84,000 paths and a total of 88 million data points. These data are collected at one-second intervals during driver service deliveries. Figure 2 presents an example of the GPS data collected in Jakarta, Indonesia. The second dataset consists of Sentinel-2A satellite imagery obtained from the United States Geological Survey (USGS) Earth Explorer. The data are presented in Figure 3 below.

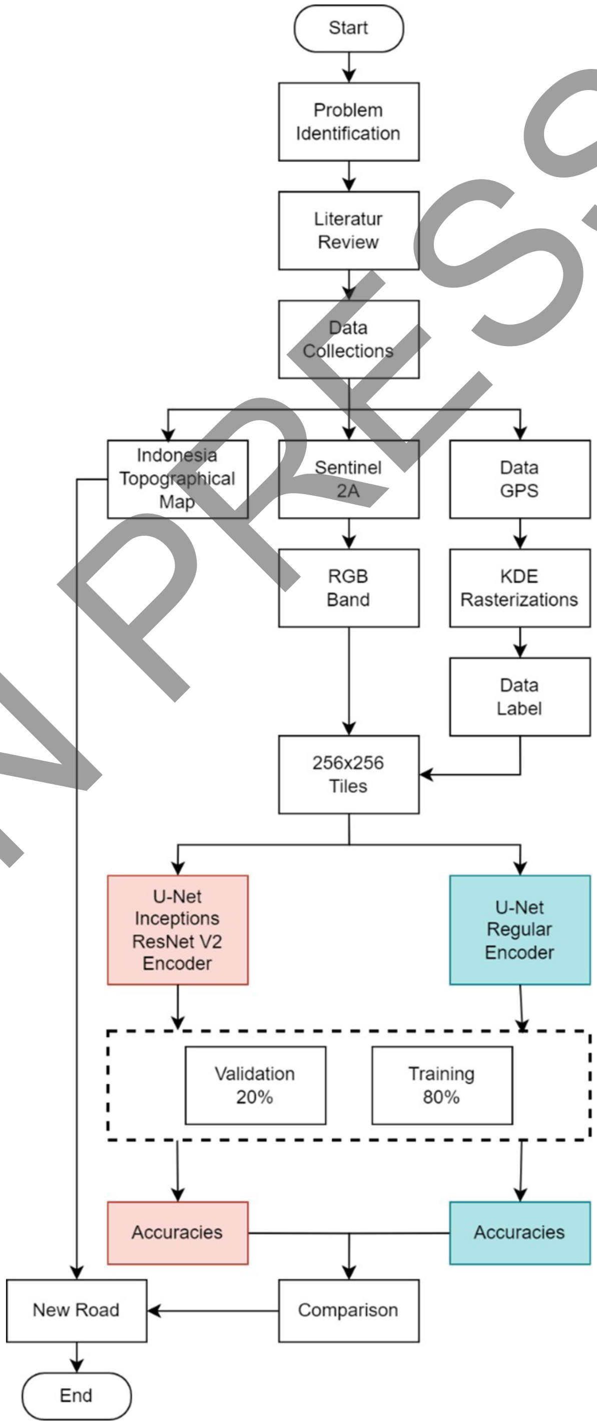


Figure 1 Research Flow

Data processing is conducted to obtain training data for road network segmentation. GPS data is used to gather labeled information related to the road network traversed by Grab drivers, while Sentinel-2A satellite imagery is utilized as the training data. The process of integrating these two data sources is illustrated in Figure 2 below. To convert the GPS point data into raster data suitable for training a satellite image segmentation model, the Kernel Density Estimation (KDE) method is applied. This method estimates the density distribution of the GPS points. The process begins with collecting GPS data from Grab drivers, which includes location points where vehicles frequently pass. Next, KDE is employed to estimate the density of these points by applying probability weights around each GPS location, resulting in a continuous representation of traffic intensity. The outcome of this estimation is then rasterized into a pixel grid, which reflects the density of vehicle movement in a specific area. Finally, this rasterized data is transformed into road labels, serving as the ground truth for training the segmentation model.

Once the road label data is obtained, the next step is to prepare Sentinel-2A satellite imagery as training data. The satellite images, which cover the same area as the GPS data, are processed using a masking technique to ensure that only the relevant parts corresponding to the roads are utilized. Next, both the satellite imagery and the road labels, in raster format, are divided into 256 x 256 pixel tiles to facilitate the model training process. With these paired satellite images and road labels, the segmentation model can be trained to recognize and map road networks from the satellite imagery automatically. This model can later be applied for various mapping and navigation purposes.

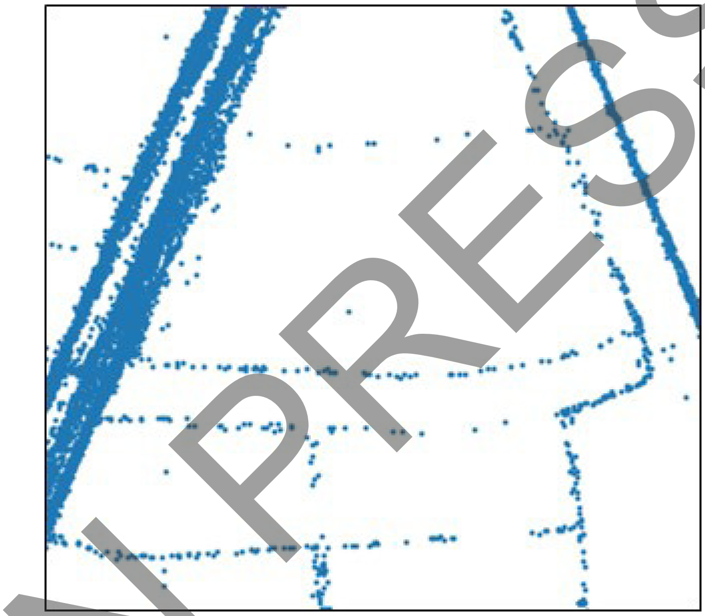


Figure 2 GPS Data from Ride-Hailing Driver Ping

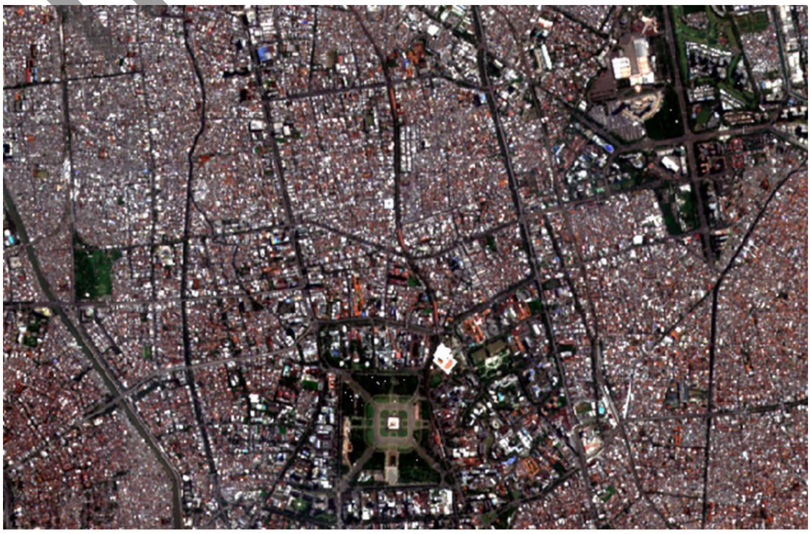


Figure 3 Sentinel 2A Satellite Imagery (RGB)

In the segmentation process, annotation or label data is necessary to train the model. These labels should accurately identify road pixels within the images. There are various methods for creating labels for road segmentation, with the most common approach being manual interpretation and digitization, which involves drawing directly on the imagery.

The method used in this research utilizes GPS data as the labeling method. The purpose of using GPS data is to ensure high accuracy, as it serves as ground truth for the road network, collected from the activities of ride-hailing drivers. By combining GPS data with satellite imagery, a high-precision model can be created, as GPS data enables the generation of detailed road maps, illustrated in Figure 4 above.

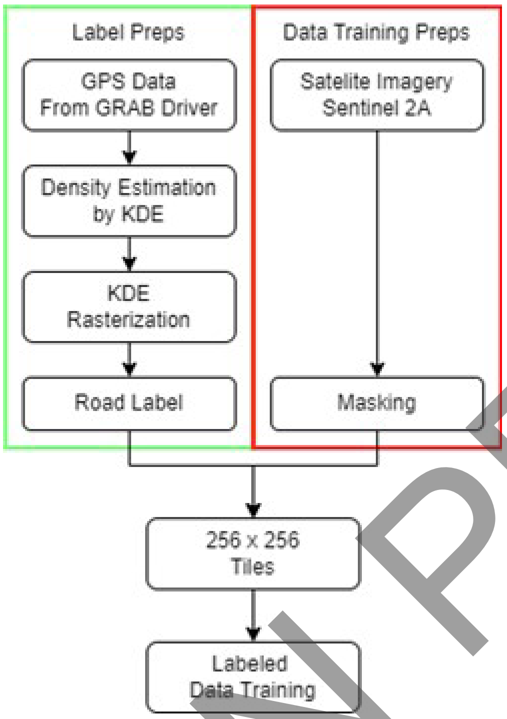


Figure 4 The Processing Flow of Data to Create the Label for the Training

KDE is a non-parametric method for estimating a density function. This approach is also used in non-parametric regression to estimate the regression function (RF). However, the choice of bandwidth plays a significant role in determining the regression outcomes. Research shows that very small or huge bandwidths lead to similar and constant estimations, respectively (Wang et al., 2022).

This research utilizes the KDE algorithm to generate a road network based on GPS ping density. The algorithm considers transportation activities and pathway patterns in the specified locations. It is represented by the following formula in Equation (1), where K = the kernel, h > 0 = the bandwidth for smoothing parameters, n = the number of data points, and h = the bandwidth (Kamalov, 2020). The KDE algorithm is commonly used for analyzing line or point data. However, this research focuses specifically on the use of point data. As a result, the conversion process generated a raster map that highlights density information, prioritizing the depiction of the road network pattern, as illustrated in Figure 5 below.

$$p_n(x) = \frac{1}{nh}\sum_{i=1}^{n} K\,(\frac{X_i - x}{h}) \qquad (1)$$

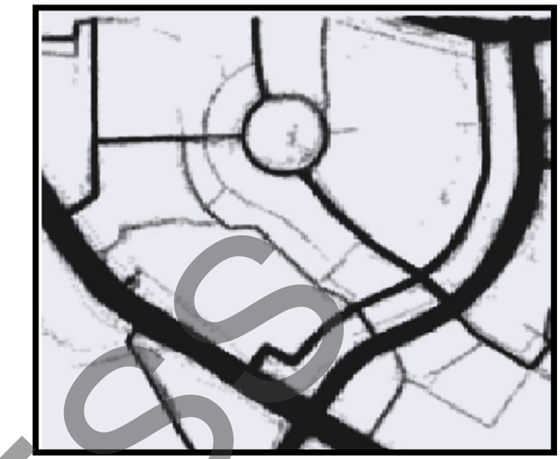


Figure 5 Kernel Density Generated by KDE

U-Net is an artificial neural network architecture designed explicitly for image segmentation tasks, which aim to distinguish objects from their background. This research employs the U-Net architecture due to its tailored design for image segmentation, which is expected to yield accurate and precise results. The U-Net architecture features a symmetrical structure composed of an encoder and a decoder, enhancing its effectiveness for segmentation tasks. The convolutional processes applied to relevant pixels significantly improve the performance of U-Net compared to other CNN architectures (Tuli et al., 2021). Figure 6 below illustrates the positions of the encoder and decoder within the U-Net architecture.

This research aims to compare the results produced by different encoders, as illustrated in Figure 6. The first encoder used is the original encoder from the U-Net architecture, which is compared to the second encoder derived from Inception-ResNet-V2. The encoder plays a crucial role in the U-Net architecture, as it is responsible for the recognition phase of the segmentation process. This phase converts the input into a feature-based representation based on the pixel values of the input (Ramba, 2020).

Inception-ResNet-V2 is an architecture developed by Google in 2016 that integrates two well-known frameworks: Inception and ResNet. The objective of using this architecture in the encoder phase is to capitalize on the strengths of Inception-ResNet-V2 for feature recognition while preserving U-Net's capabilities for accurate segmentation. The design of Inception-ResNet-V2 employs Inception-

ResNet blocks, which combine various components using skip connection networks. These skip connections between convolutional blocks are vital as they carry critical features that may be lost during the Max Pooling process, thereby helping to mitigate the vanishing gradient problem. The application of Inception-ResNet-V2 in the decoder phase is illustrated in Figure 7 below.

In Figure 7, the Inception-ResNet-V2 model plays a pivotal role by transmitting essential features to the subsequent convolutional block, even after certain aspects of the features have already been passed forward through the Max Pooling process. This dual transmission mechanism ensures that critical feature representations are preserved, thereby improving both the accuracy and completeness of the segmentation task. Nevertheless, the added computational layers increase model complexity, which may lead to slower processing speeds compared to the original U-Net model.

The first step in training the data is to create a dataset that includes both images and their corresponding labels. In this research, the training dataset comprises 2,240 images and their corresponding labels. The size of each image in the training data is 256x256 pixels, which is neither too large nor too small. This size maintains an appropriate aspect ratio of approximately 1:1, ensuring that the

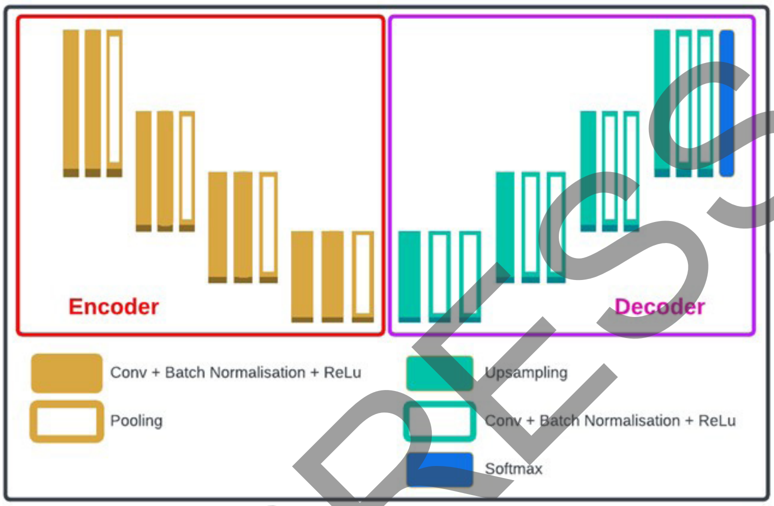

Encoder
Decoder
Conv + Batch Normalisation + ReLu
Pooling
Upsampling
Conv + Batch Normalisation + ReLu
Softmax

Figure 6 U-Net Architecture

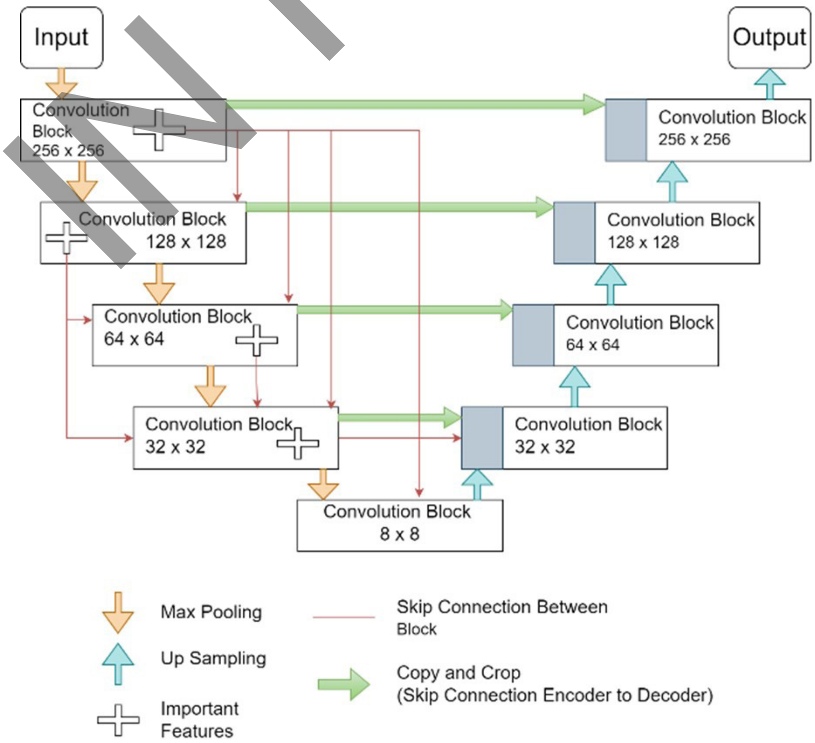

Input
Output
Convolution Block 256 x 256
Convolution Block 128 x 128
Convolution Block 64 x 64
Convolution Block 32 x 32
Convolution Block 8 x 8
Convolution Block 256 x 256
Convolution Block 128 x 128
Convolution Block 64 x 64
Convolution Block 32 x 32
Max Pooling
Up Sampling
Important Features
Skip Connection Between Block
Copy and Crop (Skip Connection Encoder to Decoder)

Figure 7 Inception-ResNet-V2 As U-Net Encoder

images retain their proportions without distortion. Figures 8 and 9 below showcase the sample dataset and the associated data labels used in this research.

Figure 8 Satellite Imagery Data for Training

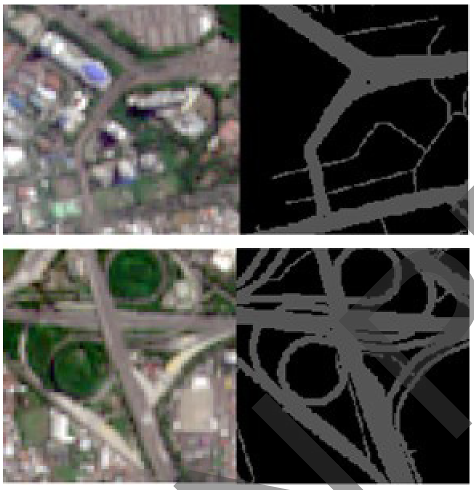


Figure 9 Satellite Imagery Data and the Label

Figure 9 shows that the road is clearly visible in the satellite imagery, and the GPS data aligns closely with the road pattern depicted in the imagery. The segmentation training will utilize the imagery data, with GPS data serving as the labels. In this research, the training data is composed of 80% for training and 20% for validation. Table 1 below presents the details of the training data composition.

Table 1 Table Data Training Compositions

Type	%	Total
Training	80	1,792
Validation	20	448

The proper composition of training and validation in deep learning is essential for minimizing the risks of overfitting and underfitting, as well as for objectively evaluating the model's performance. As previously explained, the training will utilize two different encoders: the original U-Net encoder and the Inception-ResNet-V2 encoder. Both the U-Net with the original encoder and the Inception-ResNet-V2 encoder will be trained using the same datasets. The steps involved in the training and model development are illustrated in Figure 10 below.

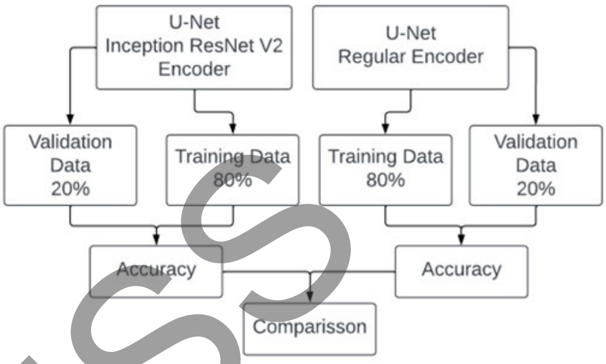


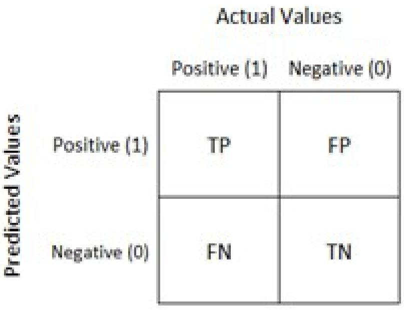
Figure 10 Training and Model Development

Once training and model development are complete, the next step is to evaluate the model using the validation datasets. This evaluation employs the Dice Loss method and a confusion matrix. The Dice Loss is a metric used to measure the overlap between two datasets; in this context, it compares the segmentation results to the labels. The Dice coefficient, also referred to as the F1-score, combines precision and recall into a single metric to reduce discrepancies between the prediction results and the ground truth (i.e., the label) during the segmentation process (Zhao et al., 2020). The Equation (2) below presents the formula for calculating Dice Loss.

$$dice\ loss = \frac{1 - (2 * intersection)}{(pred\_size + true\_size)} \quad (2)$$

The other method used to evaluate the model is the confusion matrix, which illustrates the model's accuracy in segmenting the provided image. A confusion matrix consists of four main components: True Positive (TP), which represents the number of instances the model correctly predicts as positive; True Negative (TN), which represents the number of instances the model correctly predicts as negative; False Positive (FP), which refers to the instances the model incorrectly predicts as positive; and False Negative (FN), which refers to the instances the model incorrectly predicts as negative (Pommé et al., 2022). Equation (3) shows the formula for calculating the model's accuracy using the confusion matrix. Figure 11 illustrates the concept of the confusion matrix.

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



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Figure 11 Confusion Matrix Concept

Dice Loss and the Confusion Matrix are vital tools for evaluating segmentation models, each offering unique advantages. Dice Loss, which is based on the Dice Similarity Coefficient (DSC), is particularly effective in addressing the class imbalances commonly encountered in segmentation tasks. Unlike traditional loss functions, such as Cross-Entropy Loss, which the prevalence of background pixels can skew, Dice Loss focuses directly on optimizing the spatial overlap between predicted and ground-truth masks. This approach ensures that the model learns to maximize alignment with actual segmentations, resulting in more accurate predictions, particularly for small or underrepresented regions. Moreover, Dice Loss provides a smooth gradient flow, contributing to improved model convergence and stability during training.

The Confusion Matrix provides a detailed breakdown of segmentation performance by categorizing predictions into True Positives, False Positives, False Negatives, and True Negatives. This breakdown facilitates a deeper understanding of model errors and enables the calculation of key metrics, including Precision, Recall, F1-score, and Intersection over Union (IoU). By analyzing the Confusion Matrix, practitioners can pinpoint specific areas where the model struggles, such as false detections or missed segmentations, and make the necessary improvements. Combining Dice Loss during training with the Confusion Matrix for evaluation offers a comprehensive approach to optimizing segmentation models, ensuring both accuracy and interpretability.

III. RESULTS AND DISCUSSIONS

The model training is conducted over 100 epochs for each method. The original U-Net encoder completed all 100 epochs, while the U-Net using Inception-ResNet-V2 stopped training at the 85th epoch. This early stopping occurred because the accuracy stabilized between epochs 80 and 85. The segmentation results are presented in Figures 12 and 13 below, showcasing the U-Net with the original encoder and the U-Net with the Inception-ResNet-V2 encoder, respectively. The figures illustrate that the Inception-ResNet-V2 encoder effectively segments the road and achieves a lower Dice Loss compared to the original U-Net encoder.

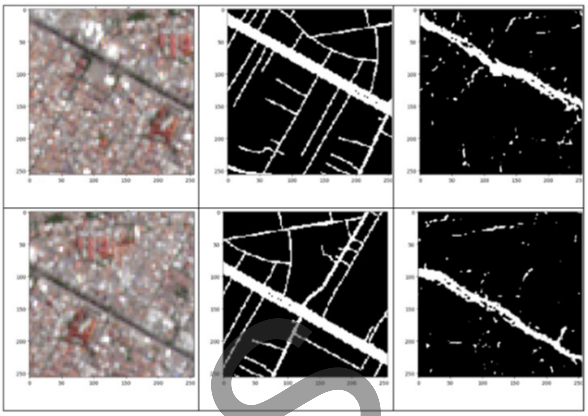


Figure 12 Segmentation Result of Classic U-Net

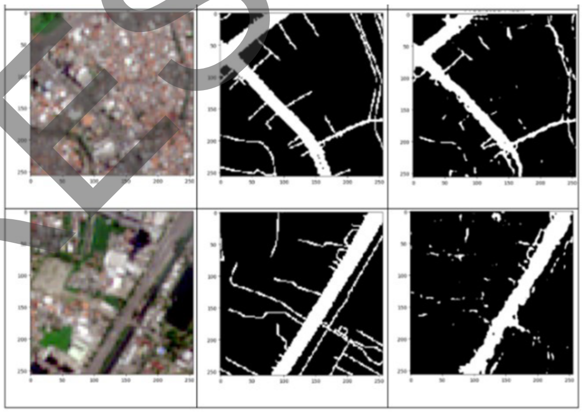


Figure 13 Segmentation Result of U-Net with Inception-ResNet Encoder

The images above compare the segmentation results of the Classic U-Net model (see Figure 12) with those of the U-Net model that uses an Inception-ResNet Encoder (see Figure 13). In Figure 12, the segmentation results from the Classic U-Net reveal some inaccuracies, particularly in detecting thinner or more complex road structures. Certain areas of the segmented map appear incomplete or exhibit noise around the edges of the roads. This indicates that the Classic U-Net may struggle to capture smaller and more intricate road features in satellite imagery.

In contrast, Figure 13, which employs the U-Net architecture with an Inception-ResNet encoder, shows significantly improved and more precise segmentation results in detecting road structures. This model proves to be more effective in capturing intricate details, especially when identifying minor roads that the Classic U-Net previously missed. The Inception-ResNet encoder likely enhances the model's ability to accurately recognize patterns, thereby reducing segmentation errors in complex areas. Overall, utilizing an advanced encoder like Inception-ResNet leads to more precise and sharper results in detecting roads from satellite images.

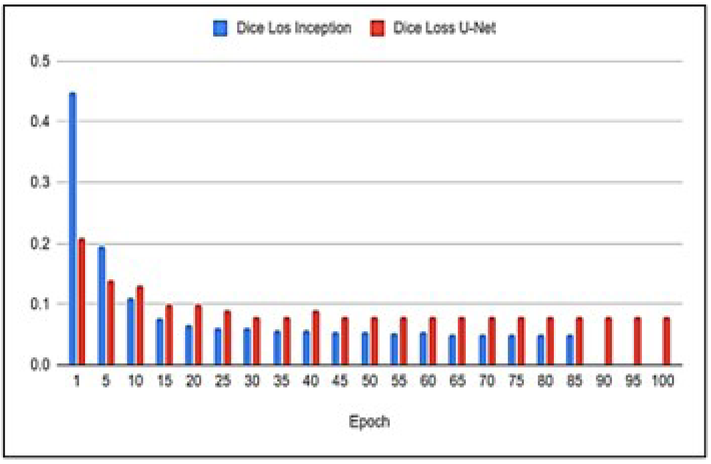


Figure 14 Dice Loss Comparison between Two Models

In Figure 14 above, it is evident that the Dice Loss generated by the CNN U-Net algorithm using the Inception-ResNet-V2 encoder is better (i.e., lower) than that of the pure U-Net encoder. Initially, the Dice Loss for the Inception-ResNet-V2 appears higher, which can be attributed to the complexity of its architecture in recognizing objects at the start of the experiment. The key difference between these two methods lies in their speed and performance. The Inception-ResNet-V2 model, due to its more sophisticated architecture, requires more training time compared to the original U-Net encoder. As shown in Figures 15 and 16 below, the Inception-ResNet-V2 encoder took eight seconds longer to train than the U-Net encoder. This additional time is due to the increased computational demands of the Inception-ResNet-V2 architecture.

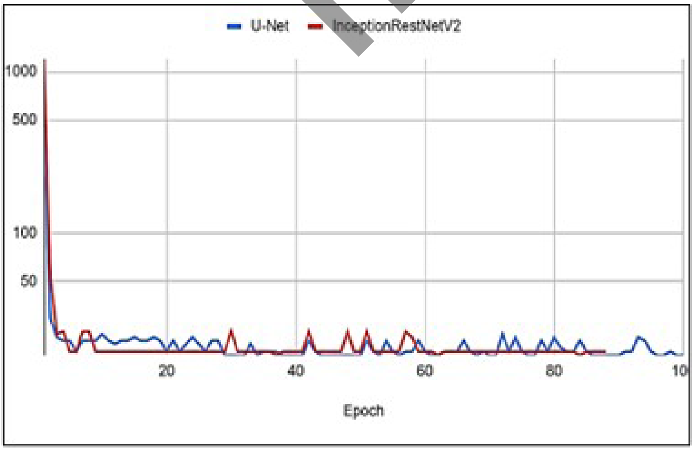


Figure 15 Computation Time on Each Epoch

In contrast, the original U-Net encoder operates more efficiently because its simpler architecture shortens the training time. Despite the difference in efficiency, both models are effective in their respective roles. However, it is essential to consider the trade-off between speed and complexity when selecting the appropriate model for a specific task.

Figure 15 illustrates the computation time per epoch for both the U-Net model and the Inception-ResNet-V2-based U-Net model. At the beginning of the training process, both models exhibit high computation times, with U-Net showing slightly more fluctuations. As training progresses, computation time stabilizes significantly, and both models demonstrate similar patterns. However, the Inception-ResNet-V2-based U-Net generally maintains a lower and more stable computation time across most epochs. This indicates that, although both models have high initial processing overhead, the Inception-ResNet-V2 architecture may offer better computational efficiency as training continues.

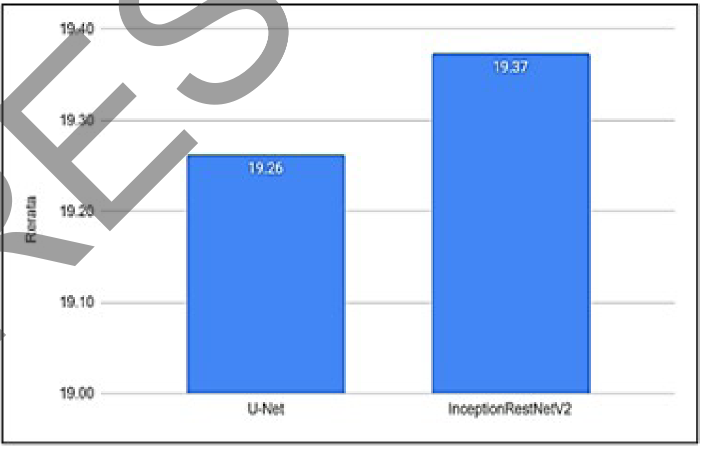


Figure 16 Average Speed Comparison

Figure 16 compares the average computation speed of the two models. The results reveal that the Inception-ResNet-V2-based U-Net has a slightly higher average speed of 19.37 compared to the Classic U-Net, which has an average speed of 19.26. Although the difference is marginal, it suggests that the Inception-ResNet-V2-based U-Net does not significantly increase processing time despite its more complex architecture. This trade-off between improved segmentation accuracy, as shown in previous figures, and a slight increase in computational speed makes the Inception-ResNet-V2 model a more efficient choice for tasks that require both high accuracy and reasonable processing time. In Table 2 below, a comparison of the two methods shows that, on average, the encoder modification using Inception-ResNet-V2 achieves better accuracy than the original encoder.

The accuracy comparison shown in Table 2 illustrates the performance differences between the Classic U-Net and the U-Net model based on Inception-

ResNet-V2. The Inception-ResNet-V2 model consistently outperforms the Classic U-Net across all key metrics, including accuracy, precision, recall, and F1 score. Specifically, the Inception-ResNet-V2 model achieves an accuracy of 91.31%, which is slightly higher than the 90.7% attained by the Classic U-Net. The improvements in precision, recall, and F1 score further indicate that the Inception-ResNet-V2 model is more effective at correctly identifying and segmenting target areas while minimizing false positives and false negatives.

Table 2 Accuracy Comparison

Type	U-Net	Inception-ResNet-V2
Accuracies	90.7%	91,31%
Precission	90.72%	91.4%
Recall	90.78%	91.2%
F1 score	90.75%	91.4%

The enhancements observed in the Inception-ResNet-V2-based U-Net can be attributed to its advanced architecture, which improves feature extraction and pattern recognition (Wang et al., 2023). The higher precision of 91.4% suggests that the model is more effective at avoiding misclassifications, while the higher recall of 91.2% indicates that it detects actual target areas more accurately. Moreover, the increased F1 score of 91.4% confirms that the model maintains a strong balance between precision and recall. These results suggest that integrating Inception-ResNet-V2 as an encoder within the U-Net structure enhances overall segmentation performance without significantly increasing computational complexity, making it a preferable choice for high-accuracy segmentation tasks.

Table 3 Accuracy Comparison with Other Research

Research	Accuracies	Model
This Research	91,31%	Inception-ResNet-V2
This Research	90.7%	U-Net
(P. Zhang et al., 2022)	87%	U-Net
(Baek et al., 2024)	87.8%	DeepLabV3+
(Baek et al., 2024)	90.5%	U-NET
(Baek et al., 2024)	93%	SIU-NET
(Wang et al., 2023)	86.28%	U-NET
(Wang et al., 2023)	89.2%	PSP-Net
(Wang et al., 2023)	84.3%	DeepLabV3+
(Wang et al., 2023)	92.19%	TransU-Net

Table 3 provides a comparative analysis of segmentation model accuracies across various research studies. This research achieved an accuracy of 91.31% using the Inception-ResNet-V2 architecture, demonstrating its superior performance compared to other models. Additionally, this research tested the U-Net model, achieving an accuracy of 90.7%, which remains competitive with existing research. Among other studies, the accuracy of U-Net varies: P. Zhang et al. (2022) reported an accuracy of 87%, while Baek et al. (2024) achieved 90.5% using the same model. DeepLabV3+, another widely used model, recorded accuracies of 87.8% (Baek et al., 2024) and 84.3% (Wang et al., 2023), indicating its effectiveness, although it performed slightly lower than Inception-ResNet-V2 in this research.

The table also includes results for SIU-NET, PSP-Net, and TransU-Net, showcasing their competitive accuracies. SIU-NET achieved the highest score among these, with 93% (Baek et al., 2024), slightly outperforming the models presented in this research. TransU-Net (Wang et al., 2023) also demonstrated strong performance at 92.19%, surpassing the U-Net-based models listed in the table. In contrast, PSP-Net, tested by Wang et al. (2023), achieved a success rate of 89.2%, indicating moderate performance. The variation in accuracies across studies highlights the influence of different architectures, datasets, and training strategies, with Inception-ResNet-V2 emerging as a strong contender for achieving high segmentation accuracy in this research.

IV. CONCLUSIONS

This research involved a segmentation process utilizing two different encoders within the U-Net architecture, which is specifically tailored for segmentation tasks. The purpose of employing these two encoders is to compare their performance and accuracy in detecting road networks using medium-resolution satellite imagery. Based on the experiments conducted for this research, the accuracies of the U-Net original encoder and the Inception-ResNet-V2 encoder are nearly identical. The Inception-ResNet-V2 encoder achieves a higher accuracy of 91%, while the U-Net original encoder has an accuracy of 90.7%. Additionally, the Dice Loss for both encoders is minimal, with only 5% for the Inception-ResNet-V2 and 8% for the original U-Net encoder. This is evident in the U-Net architecture, where either encoder effectively performs segmentation of the road network using GPS data as the labeling source. The utilization of GPS data is also beneficial, as it helps visualize the road network pattern and aids in recognizing the road network with medium-resolution imagery.

The limitation of this research is that it used only 100 epochs, which may not allow the algorithm to achieve its full potential accuracy. Some methods might perform better with more epochs, as their performance can stabilize with a higher number of iterations. Conducting further experiments with an unlimited number of epochs would be beneficial, as it would enable the algorithm to converge and stabilize fully. This approach would enable a fairer comparison between algorithms, allowing us to determine when both methods reach stability and potentially uncover performance differences that are currently hidden due to the limited number of epochs.

Further research is needed to improve the applicability and accuracy of this study. One important area that is not addressed is the measurement of the total road length in kilometers that can be extracted. Current studies should focus on comparing the extraction results from the two methods to fill this gap. By conducting this comparison, researchers can determine the total road length extracted by each method and evaluate the completeness ratio. This provides a clearer understanding of the strengths and weaknesses of each technique, leading to more reliable conclusions about their effectiveness. A more comprehensive analysis also enables this research to be applied to a broader range of fields, including urban planning and transportation management.

AUTHOR CONTRIBUTIONS

Conceived and designed the analysis; Collected the data; Performed the analysis, B. Y.; Contributed data or analysis tools; Wrote the paper, B. Y. and E. U.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, [BY], upon reasonable request. Explain the reason why the readers must request the data.

REFERENCES

Baek, W.-K., Lee, M.-J., & Jung, H.-S. (2024). Land cover classification from RGB and NIR satellite images using modified U-Net Model. *IEEE Access*, *12*, 69445–69455. https://doi.org/10.1109/ACCESS.2024.3401416

Chen, B., Ding, C., Ren, W., & Xu, G. (2021). Automatically tracking road centerlines from low-frequency GPS trajectory data. *ISPRS International Journal of Geo-Information*, *10*(3). https://doi.org/10.3390/ijgi10030122

Joubert, N., Reid, T. G. R., & Noble, F. (2020). Developments in modern GNSS and its impact on autonomous vehicle architectures. *2020 IEEE Intelligent Vehicles Symposium (IV)*, 2029–2036. https://doi.org/10.1109/IV47402.2020.9304840

Kamalov, F. (2020). Kernel density estimation based sampling for imbalanced class distribution. *Information Sciences*, *512*, 1192–1201. https://doi.org/https://doi.org/10.1016/j.ins.2019.10.017

Liu, Y., Jia, R., Ye, J., & Qu, X. (2022). How machine learning informs ride-hailing services: A survey. *Communications in Transportation Research*, *2*, 100075. https://doi.org/https://doi.org/10.1016/j.commtr.2022.100075

Luthfil, P., Program Doktor, H., Wasanta, T., & Santosa, W. (2021). Pengaruh indeks infrastruktur jalan terhadap indikator ekonomi di Indonesia. *Jurnal HPJI (Himpunan Pengembangan Jalan Indonesia)*, *7*(2), 143-152.

Pinuji, S., Savitri, A. I., Noormasari, M., Wijaya, D. K., & Kurniawan, A. (2019). Efektivitas data spasial peta Rupa Bumi Indonesia (RBI) dan Openstreetmap dalam pengambilan keputusan menggunakan Inasafe. *Jurnal Dialog Penanggulangan Bencana*, *10*(1), 22-29. https://api.semanticscholar.org/CorpusID:238144783

Pommé, L.-E., Bourqui, R., Giot, R., & Auber, D. (2022). Relative Confusion Matrix: Efficient comparison of decision models. *2022 26th International Conference Information Visualisation (IV)*, 98–103. https://doi.org/10.1109/IV56949.2022.00025

Ramba, L. S. (2020). Design of a voice controlled home automation system using Deep Learning Convolutional Neural Network (DL-CNN). *Telekontran : Jurnal Ilmiah Telekomunikasi, Kendali Dan Elektronika Terapan*, *8*(1), 57–73. https://doi.org/10.34010/telekontran.v8i1.3078

Sun, T., Di, Z., Che, P., Liu, C., & Wang, Y. (2019). Leveraging crowdsourced GPS data for road extraction from aerial imagery. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7501–7510. https://doi.org/10.1109/CVPR.2019.00769

Tuli, T. B., Kohl, L., Chala, S. A., Manns, M., & Ansari, F. (2021). Knowledge-based digital twin for predicting interactions in human-robot collaboration. *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 1–8. https://doi.org/10.1109/ETFA45728.2021.9613342

Wang, J., Liu, Y., & Chang, J. (2022). An improved model for Kernel Density estimation based on Quadtree and Quasi-Interpolation. *Mathematics*, *10*(14). https://doi.org/10.3390/math10142402

Wang, R., Cai, M., Xia, Z., & Zhou, Z. (2023). Remote sensing image road segmentation method integrating CNN-Transformer and UNet. *IEEE Access*, *11*, 144446–144455. https://doi.org/10.1109/ACCESS.2023.3344797

Wei, Y., Zhang, K., & Ji, S. (2019). Road network extraction from satellite images using CNN based segmentation and tracing. *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, 3923–3926. https://doi.org/10.1109/IGARSS.2019.8898565

Xu, Z., Yin, Y., Dai, C., Huang, X., Kudali, R., Foflia, J., Wang, G., & Zimmermann, R. (2020). Grab-Posisi-L: A labelled GPS trajectory dataset for map matching in Southeast Asia. *Proceedings of the 28th International Conference on Advances in*

Geographic Information Systems, 171–174. https://doi.org/10.1145/3397536.3422218

Zhang, J., Hu, Q., Li, J., & Ai, M. (2021). Learning from GPS trajectories of floating car for CNN-based urban road extraction with high-resolution satellite imagery. *IEEE Transactions on Geoscience and Remote Sensing*, *59*(3), 1836–1847. https://doi.org/10.1109/TGRS.2020.3003425

Zhang, P., He, H., Wang, Y., Liu, Y., Lin, H., Guo, L., & Yang, W. (2022). 3D urban buildings extraction based on airborne LiDAR and Photogrammetric Point Cloud Fusion according to U-Net deep learning model segmentation. *IEEE Access*, *10*, 20889–20897. https://doi.org/10.1109/ACCESS.2022.3152744

Zhao, R., Qian, B., Zhang, X., Li, Y., Wei, R., Liu, Y., & Pan, Y. (2020). Rethinking Dice Loss for Medical Image Segmentation. *2020 IEEE International Conference on Data Mining (ICDM)*, 851–860. https://doi.org/10.1109/ICDM50108.2020.00094