Optimized Secure Landing Identification for Unmanned Aircraft Systems

BATCH-B (GROUP-3)

1st Santhosh M
Undergraduate Student, Department of Artificial Intelligence
Amrita School of Computing, Amrita Vishwa Vidyapeetham
Coimbatore, India
CB.ENU4AIE22152

2nd Nilavarasan M
Undergraduate Student, Department of Artificial Intelligence
Amrita School of Computing, Amrita Vishwa Vidyapeetham
Coimbatore, India
CB.ENU4AIE22139

3rd L.S. Thoshi Babu
Undergraduate Student, Department of Artificial Intelligence
Amrita School of Computing, Amrita Vishwa Vidyapeetham
Coimbatore, India
CB.ENU4AIE22129

4th Deepan S
Undergraduate Student, Department of Artificial Intelligence
Amrita School of Computing, Amrita Vishwa Vidyapeetham
Coimbatore, India
CB.ENU4AIE22117

5th Kishor S
Undergraduate Student, Department of Artificial Intelligence
Amrita School of Computing, Amrita Vishwa Vidyapeetham
Coimbatore, India
CB.ENU4AIE22128

This study presents a deep learning-based approach for identifying safe landing zones for unmanned aerial systems through the segmentation of aerial imagery. Utilizing models such as UNet and PSPNet on the Semantic Drone Dataset, the research achieves precise classification of terrain types to effectively distinguish between safe and unsafe landing areas.

I. Abstract

A. *Subject*

This project presents a novel approach to autonomous drone landing zone detection through semantic segmentation of aerial images. We leverage deep learning models, specifically UNet and PSPNet, with ResNet as the encoder backbone, to accurately classify terrain types from the Semantic Drone Dataset. By fine-tuning ResNet within both architectures, we significantly improve model performance in identifying safe landing zones while effectively avoiding obstacles. Our results demonstrate that fine-tuned ResNet-based UNet outperform other configurations, achieving optimal accuracy in terrain classification for drone landing site detection.

B. *Methodologies*

We utilize deep learning models, specifically UNet and PSPNet, with a ResNet encoder for feature extraction. These models are trained on the Semantic Drone Dataset, which provides diverse aerial imagery with labeled terrain types.

C. *Proposed Result*

Our method successfully identifies optimal landing spots by classifying different terrain types. The models accurately distinguish safe landing zones from obstacles, offering reliable drone navigation solutions.

D. *Impact on Society*

The proposed approach contributes to the development of autonomous drone systems, enhancing safety and efficiency in drone operations for applications like delivery, search and rescue, and surveillance. By automating the identification of landing zones, it reduces human intervention and ensures safe landing in complex environments.

II. Introduction

In recent years, drones have gained widespread adoption across various domains, including surveillance, package delivery, and search-and-rescue operations. However, enabling drones to safely and autonomously navigate complex environments, particularly urban areas, remains a significant challenge. A crucial aspect of this autonomy is the ability to accurately assess the surroundings, including the identification of various terrain types and obstacles from an aerial perspective. Ensuring that drones can make informed decisions about where to land safely is essential for reliable and efficient operation.

This project addresses this challenge by developing a deep learning-based system that can autonomously detect optimal landing zones using aerial imagery. By leveraging semantic segmentation techniques, the model classifies terrain types and identifies areas suitable for safe landing while avoiding hazards such as buildings, trees, or other obstacles.

III. Dataset Characteristics and Composition

The Semantic Drone Dataset is designed to enhance the safety of autonomous drone landings in urban environments. It contains high-resolution aerial images of over 20 houses, captured from altitudes between 5 and 30 meters, with each image having a resolution of 6000x4000 pixels (24MP). The dataset is divided into a training set of 400 publicly available images, each annotated with segmentation masks for 24 distinct classes, such as paved areas, vehicles, vegetation, and obstacles. This dataset aids in identifying safe landing zones and avoiding hazards in diverse urban terrains.
Below image displays two urban scenes alongside their corresponding segmentation masks, showcasing the dataset's capacity for detailed and precise annotations.
In the top pair, the left image shows a paved urban environment with various elements, including pathways,

greenery, and small structures. The segmentation mask on the right accurately highlights these features, with distinct colors assigned to each class, such as vegetation, pavement, and buildings. This detailed segmentation reflects the dataset's ability to capture fine-grained details within an urban setting.

The bottom pair presents a curved road bordered by residential buildings and vegetation. The mask on the right reveals precise segmentation of each element, such as the road, vegetation, and buildings, represented by different colors. This pair emphasizes the dataset's accuracy in distinguishing complex elements in urban landscapes.

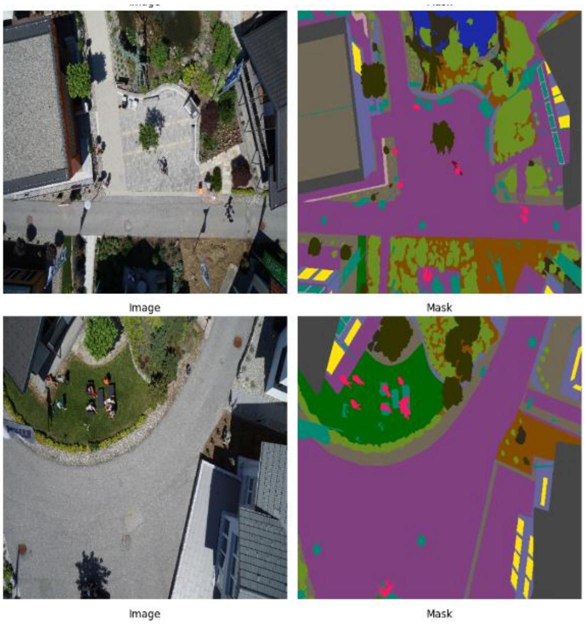

Fig. 1. Sample images from the training set with their corresponding ground truth masks.

The second image is more complex and includes a water feature with multiple types of vegetation as well as a large flag, which is segmented as an obstacle. The class distribution of the second image can be seen in Figure 2.

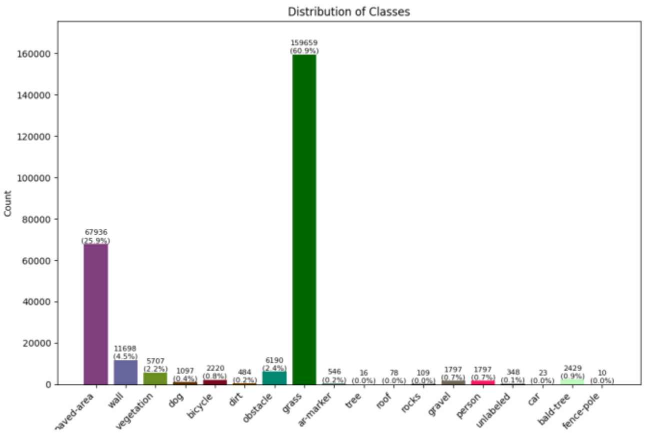

Fig. 2. Class distribution of the image's in Figure 1.

Recent The bar chart presents the distribution of various classes within the dataset, detailing the occurrence of each class by pixel count. The most prominent class is "grass," with 154,619 pixels, representing a significant portion of the dataset and indicating a strong focus on natural surfaces. The "gravel" class follows with 47,856 pixels, while "paved area" and "obstacle" also have notable representations, at 13,489 and 12,342 pixels, respectively. These values underscore the dataset's emphasis on capturing varied ground textures and barriers typically found in urban or semi-urban scenes. Other classes, such as "vehicle," "person," and "tree," have relatively lower pixel counts, suggesting they are less common within the dataset. Minor classes like "road," "fence," and "car" have the lowest representation, indicating their rare appearance.

Overall, this distribution reflects a balanced dataset, focused on diverse elements that range from natural features to constructed obstacles, providing a comprehensive overview of typical urban and natural environments

Following the visualization of dataset scenarios, the next step involves loading images and masks, along with class labels and RGB values. The dataset is split into training, validation, and test sets to ensure a thorough model evaluation. A custom dataset class and data loaders are implemented for efficient batch processing, with images processed in batches of 16. Images and masks are resized to 256x256 pixels, converted to RGB, and normalized for consistency.

To enhance model generalization, data augmentation is applied to the training set using the Albumentations library. Key augmentations include horizontal flip, vertical flip, and random 90-degree rotation, each applied with a 0.5 probability. These transformations add variability to the data, helping prevent overfitting and making the model more robust. Figure 3 illustrates examples of these spatial transformations.

IV. Methodologies

To address the challenge of semantic segmentation in drone imagery for autonomous landing detection, this study leverages two advanced deep learning architectures: UNet and PSPNet. Known for their efficacy in pixel-level classification tasks, these models are well-suited to the precise identification of landing zones within complex aerial scenes. The UNet model, with its encoder-decoder structure and skip connections, facilitates retention of critical spatial details, enabling accurate differentiation between safe landing zones and other terrain features. PSPNet, equipped with a pyramid pooling module, effectively captures multi-scale contextual information, enhancing the model's ability to interpret both local and global scene elements necessary for obstacle recognition. Both models are fine-tuned with ResNet as the backbone encoder, benefiting from pre-trained weights that improve feature extraction and classification accuracy, thus optimizing the overall effectiveness in detecting suitable landing zones.
We will then explore fine-tuned versions of the classical UNet and PSPNet [4] for comparison. Both of these pretrained models represent state-of-the-art approaches in semantic segmentation and will be implemented using the Segmentation Models PyTorch library [5].

A. UNet

UNet is a fully convolutional network designed for image segmentation. It features an encoder-decoder structure with skip connections, which helps capture both low-level and high-level features for precise segmentation. The encoder reduces spatial dimensions and extracts features, while the decoder upsamples these features to reconstruct the segmentation map.

The key components of the UNet include DoubleConvBlocks for feature extraction, DownsampleBlocks in the encoder for spatial reduction, and UpsampleBlocks in the decoder for spatial expansion. Skip connections from the encoder are concatenated with the decoder's upsampled features to retain detailed information. The network ends with a Final Convolution Layer that outputs the segmentation map with the desired number of classes.

The model has 34,526,616 parameters and handles 256x256 input images. It is trained using the Adam optimizer with a learning rate of 0.0001 and CrossEntropyLoss as the loss function, all implemented in PyTorch.

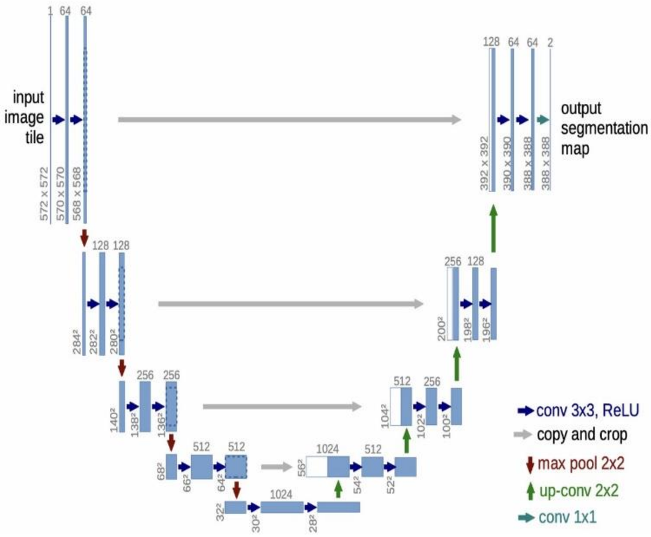

Fig. 3. Unet – Architecture

B. Fine-tuned UNet(ResNet)

ResNet (Residual Network) is a deep convolutional neural network architecture known for its innovative use of residual connections (or skip connections). These connections help address the problem of vanishing gradients, allowing very deep networks to learn effectively without degrading in performance as layers increase. ResNet achieves this by learning the “residual” (the difference between the input and output of a block), making it easier for the network to optimize deeper layers.

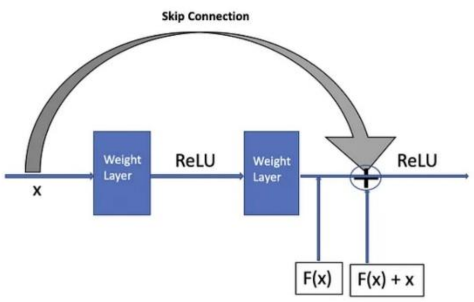

Fig. 4. Simple Residual block

ResNet variants, such as ResNet-50, are named based on the number of layers (e.g., ResNet-50 has 50 layers). ResNet-50 specifically uses four main stages with a combination of convolutional, batch normalization, and ReLU layers. This version allows for feature extraction from complex datasets and is commonly used as a backbone in transfer learning models.

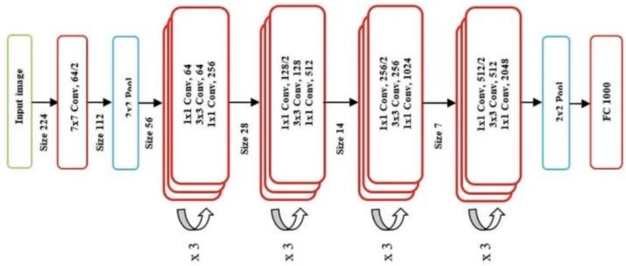

Fig. 5. ResNet - 50 architecture

C. Fine-tuned PSPNet (ResNet)

The PSPNet (Pyramid Scene Parsing Network) architecture for semantic segmentation is designed with a ResNet-50 backbone as the encoder and a pyramid pooling module for capturing global context. In this architecture, the ResNet-50 backbone extracts hierarchical features from the input image, progressively increasing the feature depth while maintaining high spatial resolution using dilated (atrous) convolutions in its final layers. These dilated convolutions, with dilation factors of 2 and 4 in the last two blocks, expand the receptive field without reducing the feature map’s resolution, allowing the network to retain spatial details crucial for pixel-level predictions.

Following feature extraction, the pyramid pooling module aggregates multi-scale context by pooling the feature map at different scales (e.g., 1x1, 3x3, 6x6, and original size), applying 1x1 convolutions to reduce depth, then upsampling and concatenating these pooled features. This fusion of multi-scale context enhances the model's ability to classify each pixel accurately based on both local and global information. In the decoding stage, an 8x bilinear upsampling or a U-Net-style decoder is often used to produce the final high-resolution segmentation map. This combined architecture makes PSPNet highly effective at capturing both fine and broad details, essential for precise semantic segmentation.

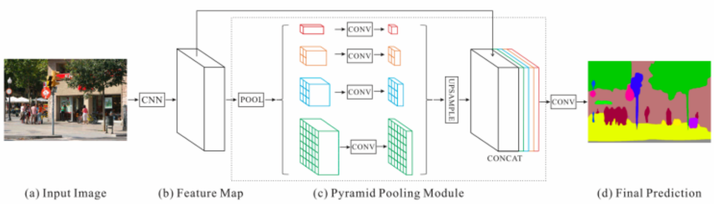

Fig. 6. PSPNet-Architecture

V. Results and Discussion

The training of all models was performed over 100 epochs, during this process, the train and validation loss and accu racy were monitored to track the models’ performance and convergence.

Model	Train Loss	Train Accuracy
UNET	0.6088	0.8275
UNET - Fine Tuned	0.2791	0.8898
PSPNET - Fine Tuned	0.3374	0.8427

Table.1. Train Loss

Model	Val Loss	Val Accuracy
UNET	0.7482	0.7911
Unet - Fine Tuned	0.3595	0.8599
PSPNET - Fine Tuned	0.4236	0.8180

Table.2. Validation Loss

Model	Test Loss	Test Accuracy
UNET	0.7829	0.7935
Unet - Fine Tuned	0.3275	0.8532
PSPNET - Fine Tuned	0.3774	0.8113

Table.3. Test Loss

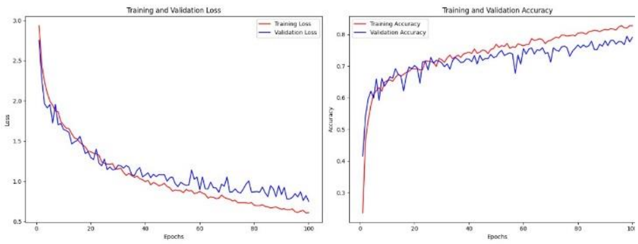

Fig. 7. Training And Validation Loss Of UNet

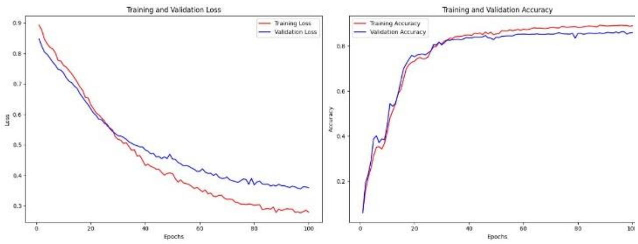

Fig. 8. Training And Validation Loss Of Unet - Fine Tuned

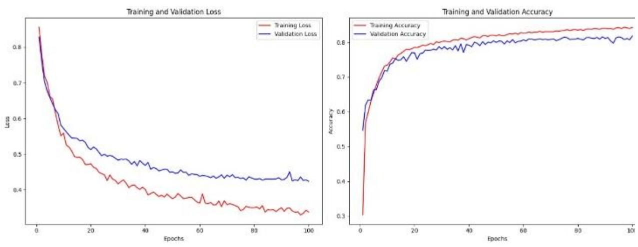

Fig. 9. Training And Validation Loss Of PSPNet

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting. Please note sections A-D below for more information on proofreading, spelling and grammar.

VI. LITERATURE SURVEY

A. ***Research:*** *Semantic Drone Dataset [1]*

Dataset: Semantic Drone Dataset
Unique Contributions: Provides high-resolution drone images annotated with 24 classes, focusing on urban and natural terrains.
Our Project: Uses this dataset for model training, enabling accurate segmentation of terrain features to support safe landing detection.

B. ***Research:*** *Albumentations Library [2]*

Unique Contributions: Data augmentation library known for fast and flexible image augmentations.
Our Project: Applies spatial-level augmentations (e.g., flips, rotations) to improve model robustness, particularly effective for diverse terrain types.

C. ***Research:*** *U-Net (Ronneberger et al.) [3]*

Models Used: U-Net (designed for biomedical images)
Dataset: MICCAI 2015
Unique Contributions: Encoder-decoder architecture with skip connections, effective for pixel-level image segmentation tasks.
Our Project: Customizes U-Net with improved layers for drone applications, achieving accurate segmentations and identifying challenging classes.

D. ***Research:*** *Pyramid Scene Parsing Network (PSPNet) [4]*

Models Used: PSPNet
Unique Contributions: Aggregates contextual information across scales, excelling in complex scenes with detailed segmentation needs.
Our Project: Uses fine-tuned PSPNet to improve recognition of both local and global contexts in drone images, though with lower detail accuracy.

E. ***Research:*** *Segmentation Models PyTorch [5]*

Models Used: PyTorch U-Net, PSPNet architectures
Unique Contributions: Provides flexible pre-trained segmentation models in PyTorch, aiding in fast implementation.
Our Project: Adopts pre-trained U-Net and PSPNet from this library, allowing for transfer learning on drone-specific tasks.

F. ***Research:*** *PyTorch Library [6]*

Models Used: General Deep Learning
Unique Contributions: Optimized for high-performance deep learning, supporting various model architectures and training procedures.
Our Project: Implements models with PyTorch, tuning the training process to improve accuracy, and using Adam optimizer and Dice loss.

G. ***Research:*** *ImageNet Pre-training [7]*

Models Used: ResNet-50 Backbone Pre-training
Dataset: ImageNet
Unique Contributions: Transfer learning with pre-trained weights from ImageNet, enhancing model performance with limited data.
Our Project: Fine-tunes ResNet-50 for both U-Net and PSPNet, effectively adapting model weights to drone image segmentation tasks.

H. ***Research:*** *PASCAL VOC Challenge [8]*

Models Used: N/A
Dataset: PASCAL VOC 2012
Unique Contributions: Benchmark for object detection and segmentation, highlighting advances in visual recognition.
Our Project: Fine-tuned U-Net and PSPNet are compared to benchmarks in segmentation tasks, showing improved segmentation for urban scenes.

I. ***Research:*** *Cityscapes Dataset [9]*

Models Used: N/A
Dataset: Cityscapes
Unique Contributions: Semantic urban scene understanding with complex annotations, often referenced for model evaluation.
Our Project: Evaluates model robustness against complex urban classes, indicating areas for enhancement with additional datasets.

J. ***Research:*** *SciPy Distance Transform [10]*

Models Used: N/A
Dataset: N/A
Unique Contributions: Provides efficient algorithms for computing Euclidean distance transforms, crucial for spatial calculations in image processing.
Our Project: Employs SciPy's distance transform to calculate safe drone landing zones by assessing distance to obstacles and image boundaries.

VII. Novelity (Landing Identification For UAS)

This project also included the development of an autonomous algorithm to identify safe landing spots for drones using segmented aerial images. The algorithm distinguishes between safe and avoid classes: safe classes (e.g., paved and grassy areas) are suitable for landing, while avoid classes (e.g., roofs, trees, and people) pose potential obstacles. First, separate masks are created for both safe and avoid classes. A Euclidean distance transform is then applied, using SciPy, to compute the distance from each pixel to the nearest avoid class. This transform allows for selecting areas sufficiently distant from obstacles. Additionally, a distance transform from the image borders is applied to avoid landing too close to the edges, enhancing safety.

A composite distance map combines these transforms, identifying regions both distant from obstacles and clear of edges. The algorithm then scans for potential landing spots of a predefined size, selecting the location with the highest safety score based on the average values from the distance map. The output includes the best landing coordinates, a safety mask, and the composite distance heatmap. Observations on test images reveal accurate safety heatmaps and effective segmentation of urban features like buildings, vegetation, and objects, though the model sometimes misclassifies gravel-covered roofs as safe areas.

Despite minor segmentation inaccuracies, the fine-tuned UNet and landing algorithm consistently locate safe landing zones across test scenarios. These findings emphasize the reliability of the combined model and algorithm for safe drone landing, though further refinement may be needed for complex cases, such as gravel-covered roofs, to improve real-world robustness.

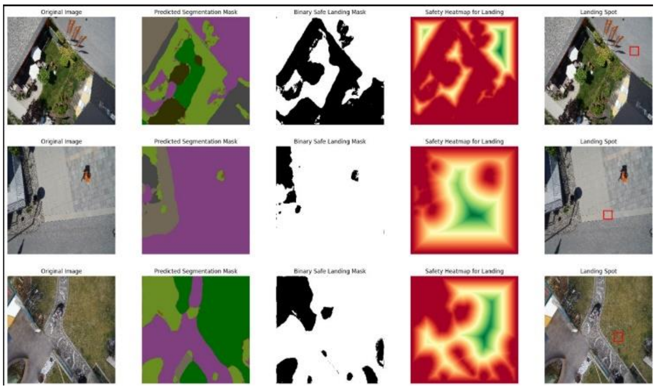

Fig. 10. Prediciting Optimal Spot For Drone To Land.

VIII. Conclusions

This project focused on enhancing scene understanding and identifying safe landing zones for drones in urban environments using deep learning and semantic segmentation. Our models were designed to label every pixel in aerial images, distinguishing between different terrains and objects to ensure safe operations. Among the architectures tested, the fine-tuned UNet demonstrated the highest accuracy with a mean Intersection over Union (mIoU) of 58.01%, particularly excelling in complex urban features. Data augmentation further improved model performance, highlighting its effectiveness in building robust segmentation models.

Qualitative analysis revealed that the fine-tuned UNet achieved the most detailed and precise segmentations, outperforming both the manually defined UNet and PSPNet. While PSPNet produced smoother outputs, it often missed finer details. The manually defined UNet, despite its lower overall metrics, occasionally produced visually consistent results. Our drone landing detection algorithm, designed to process segmented images, effectively identified safe landing spots by combining distance transform maps, avoiding obstacles and borders with high reliability.

This work uncovered areas for potential improvement, such as using higher-resolution images and more diverse data augmentation techniques. Future research could explore alternative architectures, like DeepLabV3+, and implement real-time testing on live drone footage to evaluate processing efficiency. Overall, our findings underscore the value of precise segmentation and diverse datasets in advancing autonomous drone technology for complex urban scenarios.

ACKNOWLEDGMENT

We would like to express our special thanks of gratitude to our teacher (Dr. Mithun Kumar sir), who gave us the golden opportunity to do this wonderful project on the topic ("*Optimized Secure Landing Identification for Unmanned Aircraft Systems*"), which also helped us in doing a lot of Research and we came to know about so many new things. We are thankful for the opportunity given.

REFERENCES

[1] I. of Computer Graphics and G. U. o. T. Vision, "Semantic Drone Dataset," 2019.

[2] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: Fast and Flexible Image Augmentations," *Information*, vol. 11, no. 2, 2020, doi: 10.3390/info11020125.

[3] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18, 2015, pp. 234–241.

[4] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.

[5] P. Iakubovskii, "Segmentation Models Pytorch," *GitHub*, 2019.

[6] A. Paszke et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., pp. 8024–8035, 2019. Available: PyTorch Paper.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, 2009, pp. 248–255.

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results."

[9] M. Cordts et al., "The Cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[10] P. Virtanen et al., "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020, doi: 10.1038/s41592-019-0686-2.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.