Semantic Segmentation using Deep Learning for Aerial Images

Hector Eduardo Tovanche-Picón[1], Diego Mercado Ravell[2]

[1]Department of Industrial Engineering and Manufacturing, The Autonomous University of Ciudad Juarez, Cd. Juarez, 32584, Chihuahua, Mexico

[2]Center for Research in Mathematics, Quantum Knowledge City, Zacatecas, Mexico

Abstract

In this article, a convolutional neural network model is presented for semantic segmentation of aerial images in urban areas using the VGG16 architecture as the encoder and UNet as the decoder. The model was trained and evaluated using the publicly available dataset named Semantic Drone, which consists of aerial images acquired at altitudes ranging from 5 to 30 meters. Various data augmentation techniques, such as random elastic deformation and brightness adjustment, were applied to enhance the model's generalization capability. The obtained results show an average accuracy of 81% in segmenting 23 different classes, including people, cars, and dogs. Additionally, an inference speed of up to 50 fps was achieved after optimizing the model on a GPU. Overall, the proposed model has the potential to be employed in drone security applications and decision-making processes in urban areas.

Keywords

semantic segmentation, deep learning, VGG16, UNet, data augmentation

1. Introduction

In recent years, there has been significant interest in the application of deep learning techniques for semantic segmentation of aerial images [1, 2, 3, 4, 5]. One of the most popular techniques is the use of Convolutional Neural Networks (CNNs), which have been successfully applied in various computer vision applications. CNN-based models have demonstrated a remarkable ability to learn relevant features in images and segment different object classes with high accuracy and efficiency [3, 6, 7]. Additionally, other neural network architectures such as Fully Convolutional Networks (FCNs) [8], Encoder-Decoder Networks (ENC-DEC) [9, 10, 11, 12], and Attention Networks (SAN) [13] have also shown promising results in semantic image segmentation. However, the application of these techniques in semantic segmentation of aerial images faces challenges such as variability in object appearance and texture, the presence of shadows and reflections, and the lack of labeled data. Despite these challenges, the use of deep learning techniques in semantic segmentation of aerial images remains an active and evolving research area, with numerous opportunities for developing new models and approaches to further enhance accuracy and efficiency in this critical task.

CISETC 2023: International Congress on Education and Technology in Sciences, December 04–06, 2023, Zacatecas, Mexico

hector.tovanche@uacj.mx (H. E. Tovanche-Picón); diego.mercado@cimat.mx (D. M. Ravell)

https://sites.google.com/view/ph-d-diego-mercado (D. M. Ravell)

0000-0001-5073-633X (H. E. Tovanche-Picón); 0000-0002-7416-3190 (D. M. Ravell)

In the scientific literature, different strategies have been proposed to address challenges associated with deep learning-based semantic segmentation in aerial images. For example, new techniques have been developed to generate synthetic data and augment the training dataset [14], which can help improve model generalization. Image preprocessing techniques, such as atmospheric correction [15] and image normalization [16], have also been proposed to enhance input data consistency and quality. Additionally, hybrid approaches combining deep learning techniques with traditional image processing methods have been suggested [17], leveraging the advantages of both approaches to overcome their limitations. In addition to deep learning techniques, there are other strategies for semantic segmentation of aerial images, such as feature-based and graph-based approaches [18]. Feature-based approaches focus on extracting relevant features from images, such as texture and shape, to segment different object classes [19]. A state-of-the-art review in deep learning-based semantic segmentation for aerial images reveals a broad research field with numerous promising techniques and approaches that have the potential to significantly improve the accuracy and efficiency of semantic segmentation in aerial images.

This work presents the evaluation of a convolutional neural network applied to the task of semantic segmentation of aerial images using the VGG16 and U-Net architectures with a publicly available dataset for training and validation. The remainder of the document is structured as follows: in Section 2, the selected architecture for training the semantic segmentation model, the dataset used, and data augmentation techniques are detailed. Section 3 describes in detail the selected parameters for the training stage of the model. Section 4 presents the experimental results of our work, focusing on evaluating the accuracy of the deep learning-based semantic segmentation model for aerial images on a public dataset for each of the classes and the optimization required to run the model in real-time. Finally, Section 5 discusses the conclusions of the work and presents possible future directions for research in this field.

2. Semantic Segmentation in Aerial Images

In this section, we will describe in detail an approach to semantic segmentation based on a neural network architecture that combines an encoder based on the VGG16 architecture [20] and a decoder based on the U-Net architecture [9].

The VGG16 architecture [20] represents a prominent convolutional neural network with deep significance in image classification tasks. Its structure is characterized by a sequence of convolutional and pooling layers, followed by fully connected layers at the top of the network. In the task of semantic segmentation, the VGG16 network plays a fundamental role as an encoder capable of extracting highly relevant features from input images. This architecture has earned a prominent place in the fields of computer vision and deep learning due to its ability to understand and represent complex features in images, making it valuable in a variety of applications.

On the other hand, the U-Net architecture [9] is presented as an encoder-decoder neural network designed specifically to address challenges in semantic segmentation of images, initially conceived for medical applications. However, its versatility has proven its suitability in various domains, including semantic segmentation of aerial images. This architecture is distinguished by

its dual structure, comprising a downward section that uses convolutional and pooling layers to reduce the spatial resolution of the image, followed by an upward section that uses upsampling and concatenation layers to increase spatial resolution and generate the final segmentation mask. The U-Net architecture has become an essential tool in image processing, enabling precise and detailed segmentation in a wide range of applications, from medical diagnostics to mapping land surfaces from the air.

By combining both architectures into a single architecture, see Table 1, the inherent strengths of VGG16 as an encoder and U-Net as a decoder are leveraged, resulting in a highly effective approach for semantic segmentation of images. VGG16, with its deep structure of convolutional and pooling layers, excels at extracting visually relevant features from input images, identifying patterns, textures, and key details. These features, acting as high-level knowledge, are essential for semantic segmentation. On the other hand, U-Net, with its specific encoder-decoder design, specializes in the precise reconstruction of segmentation masks. The downward section of U-Net simplifies the task by reducing spatial resolution, while the upward section recovers fine details and local context. The key to this combination lies in the seamless transition between both architectures, using features extracted by VGG16 as input for the upward section of U-Net. This approach provides accurate and consistent segmentation by combining rich detail information and local context with high-level features.

Table 1
Simplified combined architecture of VGG16 and U-Net for semantic segmentation.

Stage	Layer	Operation	Parameters
Encoder (VGG16)	Convolutional Layer	2D Convolution	64 filters, kernel 3x3, ReLU activation
		2D Convolution	64 filters, kernel 3x3, ReLU activation
		2D Max Pooling	kernel 2x2
	Convolutional Layer	2D Convolution	128 filters, kernel 3x3, ReLU activation
		2D Convolution	128 filters, kernel 3x3, ReLU activation
		2D Max Pooling	kernel 2x2
Decoder (U-Net)	Upsampling Layer	2D Deconvolution	kernel 2x2
	Concatenation Layer	Concatenation	with output from the corresponding stage of the encoder
	Convolutional Layer	2D Convolution	128 filters, kernel 3x3, ReLU activation
Output	Convolutional Layer		2D Convolution, 1 filter, kernel 1x1, Sigmoid activation

In the proposed architecture, the VGG16-based encoder is employed to extract features from the input aerial images, which are then fed into the U-Net-based decoder to generate the final segmentation mask. Additionally, regularization techniques such as dropout and batch normalization are utilized to enhance generalization and prevent overfitting.

2.1. Image Division Based on Altitude for Semantic Segmentation

The categorization of aerial images by altitude is a key approach in semantic segmentation, as the features and objects present in the images vary significantly depending on the altitude at which the image was captured.

These images can be classified into three main categories based on their acquisition altitude: low, medium, and high. Low-altitude images are typically captured at heights of less than 30 meters and show fine details of objects such as buildings, vehicles, and pedestrians. Medium-altitude images are obtained at altitudes between 30 and 150 meters, providing a broader view of the photographed area, allowing for a better understanding of the context and distribution of objects in a scene. On the other hand, high-altitude images are taken at altitudes above 150 meters and offer an overview of a region, facilitating the understanding of the distribution of objects on a large scale.

This categorization by altitude enables semantic segmentation models to focus on specific features of the images that are relevant to the corresponding acquisition altitude. This can significantly improve the accuracy and efficiency of semantic segmentation models, especially when deep learning techniques are employed.

2.2. Dataset

The dataset used in this work is the Semantic Drone Dataset [21], which focuses on the semantic understanding of urban scenes to enhance the safety of autonomous drone flight and landing procedures. The dataset's images depict more than 20 houses from a top-down (bird's-eye) view acquired at an altitude of 5 to 30 meters above the ground. A high-resolution camera was used to capture images of size 6000 × 4000px (24Mpx). This dataset includes labels for 24 different classes; Table 2 displays the 24 classes and their assigned RGB values. Figure 1 shows four examples of RGB images and the various represented scenarios. Figure 2 displays the corresponding masks for the example images, where different colors represent the classes present in the dataset. The training set consists of 400 publicly available images, while the test set comprises 200 private images. This dataset is widely used in research on deep learning-based semantic segmentation for aerial images due to the diversity of objects and urban contexts presented in the images, posing an interesting challenge for machine learning models.

2.3. Data Augmentation

Generating synthetic data or applying data augmentation techniques is a common approach to enhance the ability of semantic segmentation models to generalize and adapt to different scenarios and conditions. Data augmentation involves creating new images from the original ones by applying random transformations, such as rotation, scaling, brightness changes, contrast adjustments, among others.

A common data augmentation technique used in semantic segmentation is called "elastic deformation-based data augmentation," which involves applying a random elastic deformation to the original image, creating a new synthetic image. This is achieved by adding a small fraction of a random elastic displacement field to the original position of each pixel. The random elastic displacement field is generated by a white noise function that is turned into a vector field

Table 2
RGB Values for Dataset Classes [21]

Name	R	G	B	Name	R	G	B
Unlabeled	0	0	0	Door	254	148	12
Paved Area	128	64	128	Fence	190	153	153
Soil	130	76	0	Fence Post	153	153	153
Grass	0	102	0	Dog	102	51	0
Gravel	112	103	87	Car	9	143	150
Water	28	42	168	Bicycle	119	11	32
Rocks	48	41	30	Tree	51	51	0
Pool	0	50	89	Bare Tree	190	250	190
Vegetation	107	142	35	AR Marker	112	150	146
Roof	70	70	70	Obstacle	2	135	115
Wall	102	102	156	Conflict	255	0	0
Window	254	228	12	Person	255	22	96

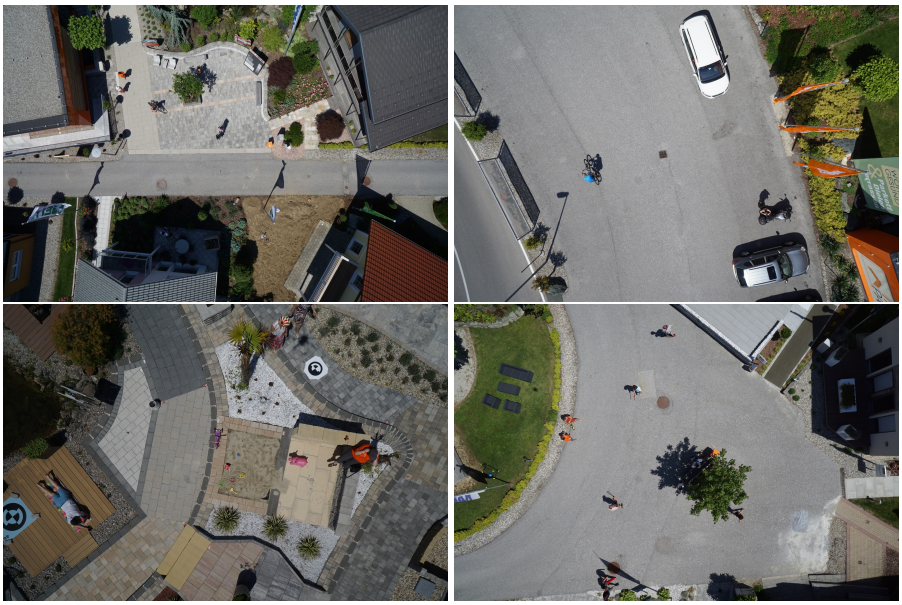

Figure 1: Examples of RGB images from the dataset [21].

through the application of a Gaussian filter. The mathematical formula for this transformation is represented in Equation 1.

$$x'_{ij} = x_{ij} + u_{ij} + \frac{\partial u_{ij}}{\partial x_i} + \frac{\partial u_{ij}}{\partial x_j} \tag{1}$$

Where x'_{ij} is the position of a pixel in a deformed image, x_{ij} is the original position of the pixel, and u_{ij} is the displacement of the pixel's position. The formula indicates that the deformed position of the pixel is equal to its original position plus the displacement, along with the

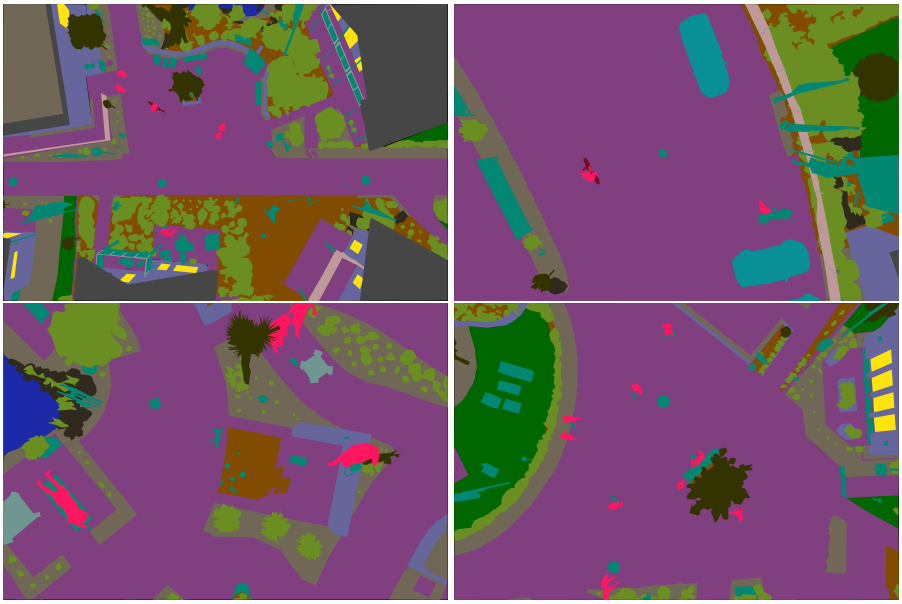

Figure 2: Examples of masks from the dataset [21].

contribution of the partial derivatives of the displacement function u in the x_i and x_j directions.

This technique is used to simulate deformations that may occur in an aerial image due to factors such as lens distortion, aerial vehicle movement, among others.

Another popular data augmentation technique is random cropping, which involves cutting a random portion of the original image and using it as a new image. This technique is particularly useful for creating synthetic images containing partially visible objects, which can help the semantic segmentation model learn to recognize objects in challenging conditions.

Equation 2 represents the random cropping function used,

$$x_i = \text{random}(0, w - p);\ y_i = \text{random}(0, h - q) \tag{2}$$

Where w and h are the width and height of the original image, respectively, and p and q are the width and height of the desired crop, random() represents a function that returns a random number within the specified range.

The data augmentation technique by brightness adjustment involves adjusting the brightness of images to enhance the model's ability to generalize and handle different lighting conditions. This technique involves adding a constant value to all pixels in the image, which increases or decreases brightness. The formula used for brightness adjustment can be seen in Equation 3.

$$I_b = I_o + v \tag{3}$$

Where $I_{original}$ is the original image, I_{bright} is the image with increased brightness, and v is the constant value added to each pixel. The value of v can be randomly generated within a specific range to create variations in the image's brightness.

All images used during the training stage undergo these data augmentation techniques to increase the size of the dataset. Figure 3 shows two results of data augmentation, and Figure 4 shows the corresponding masks after being processed with the same techniques.

Figure 3: RGB images after data augmentation techniques, Random Crop, Elastic Deformation, Brightness Change, respectively.

Combining different data augmentation techniques can significantly increase the diversity and quantity of data available for model training, improving its accuracy and ability to generalize to new scenarios and conditions. However, it is important to note that excessive data augmentation can also result in overfitting the model to the training data, limiting its ability to generalize to new data.

3. Model Training

The training process of the machine learning-based segmentation model requires certain resources and tools. One of the most crucial elements is hardware, as computational resources are needed to effectively carry out the model training. In other words, a computer with sufficient processing power is required to perform the necessary mathematical operations for training the model.

Additionally, it is necessary to define training hyperparameters, which are variables that control the model training process. These hyperparameters may include batch size, learning rate, number of epochs, among others. Proper selection of hyperparameters can have a significant impact on the model's performance.

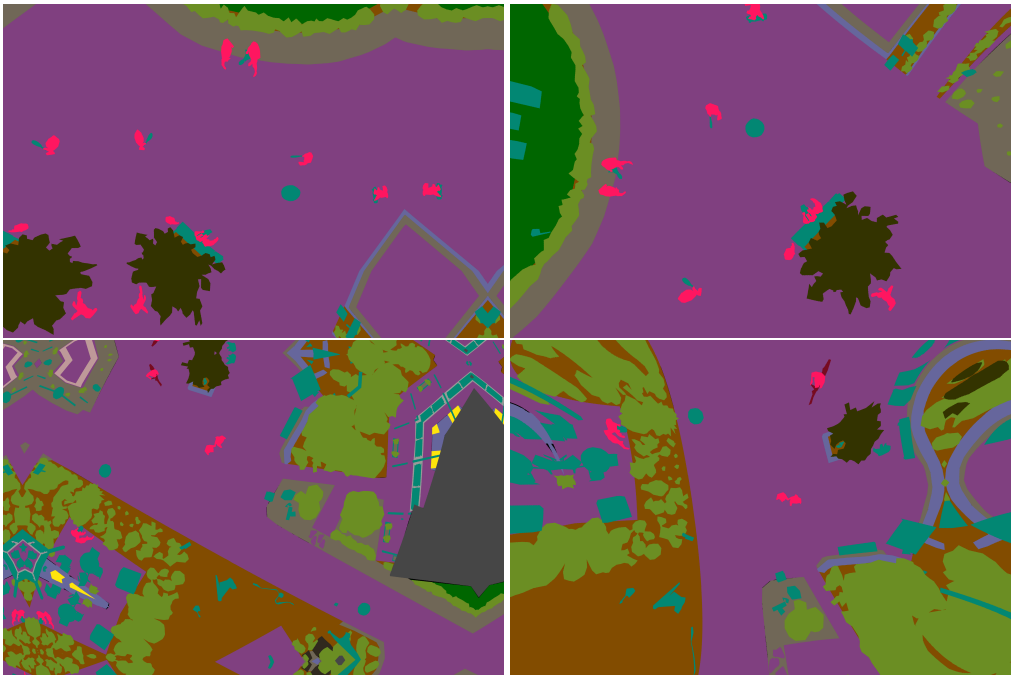

Figure 4: Masks after applying data augmentation techniques

3.1. Hardware Specifications

To conduct the training of the neural network used in this study, a computer equipped with a Windows 11 operating system, an Intel Core i7 processor, an NVIDIA GeForce RTX 2060 graphics card, and 32 GB of RAM was employed. This type of hardware configuration is commonly used in deep learning tasks due to its high processing capability and available memory.

The RTX 2060 graphics card, chosen for this study, plays a crucial role in the training process of neural networks thanks to its Tensor core architecture. This feature enables substantial acceleration in matrix operation calculations, an essential function in image processing, particularly in applications like semantic segmentation.

The extensive RAM capacity available in the computer system is an essential resource that allows efficient storage and processing of large datasets, such as the one used in this study. This enhanced capacity significantly facilitates the neural network training process and ultimately reduces the processing time required to achieve a well-performing trained model.

3.2. Training Hyperparameters

For the training of the neural network, various hyperparameters were utilized that influence the performance and accuracy of the semantic segmentation model for aerial images.

Firstly, an input size of 256×256 pixels was used for each image. This size was chosen to balance model accuracy with the time and resources required for training.

The batch size, referring to the number of images used in each iteration during training, was

set to 8. This choice was based on the available memory capacity in the hardware used for neural network training.

The number of epochs was fixed at 200, meaning the training dataset was iterated 200 times to adjust the neural network weights and optimize the model.

The initial learning rate was set to $100e - 6$, indicating the rate at which the neural network weights are updated during training. This value was adjusted to ensure an appropriate convergence rate of the model.

Lastly, a Dice coefficient function was used for calculating the stochastic gradient descent in the optimization process. This function is commonly employed as a metric for evaluating semantic segmentation models, allowing the comparison of the overlap area between the predicted segmentation mask and the ground truth segmentation mask.

4. Results

4.1. Evaluation Metrics

In order to assess the performance of the trained model, two widely used metrics in the literature are proposed: the Dice coefficient and the Jaccard coefficient or Intersection over Union.

4.1.1. Dice Coefficient

The Dice coefficient is a similarity metric employed in semantic segmentation tasks, such as deep learning-based aerial image segmentation. It is commonly used as a loss function during model training and can also be used to evaluate the quality of segmentation on the test set.

The Dice coefficient is calculated as the ratio between the area of intersection between the segmentation mask generated by the model and the true mask of the image, and the total area of the combined two masks. The Dice coefficient value ranges from 0 to 1, where a value of 1 indicates perfect segmentation—meaning the model-generated mask precisely matches the real mask of the image. A value of 0, on the other hand, indicates that the segmentation performed by the model is completely incorrect.

The Dice coefficient can be calculated using the following mathematical formula:

$$Dice = \frac{2 \times |X \cap Y|}{|X| + |Y|} \tag{4}$$

Here, X and Y are the masks of the segmentation generated by the model and the actual segmentation of the image, respectively. The symbol $\cap$ denotes the intersection operation between two sets, and $|X|$ and $|Y|$ represent the size of sets X and Y, respectively.

4.1.2. Jaccard Coefficient

The Jaccard Coefficient is a commonly used metric to assess the similarity between two datasets. In the context of semantic segmentation of aerial images, the Jaccard Coefficient can be employed to measure the accuracy of the segmentation obtained by the neural network.

The formula for the Jaccard Coefficient is expressed as the ratio between the intersection of two sets and their union, and is defined as follows:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{5}$$

Here, A and B are the sets being compared, $A \cap B$ is their intersection (i.e., the elements common to both sets), and $A \cup B$ is their union (i.e., all elements appearing in at least one of the sets).

4.2. Evaluation on Test Dataset

To assess the model's performance, a cross-validation was conducted using 100 images from the public dataset that were not part of the training set. Figure 5 presents examples of input images and predicted masks compared to ground truth.

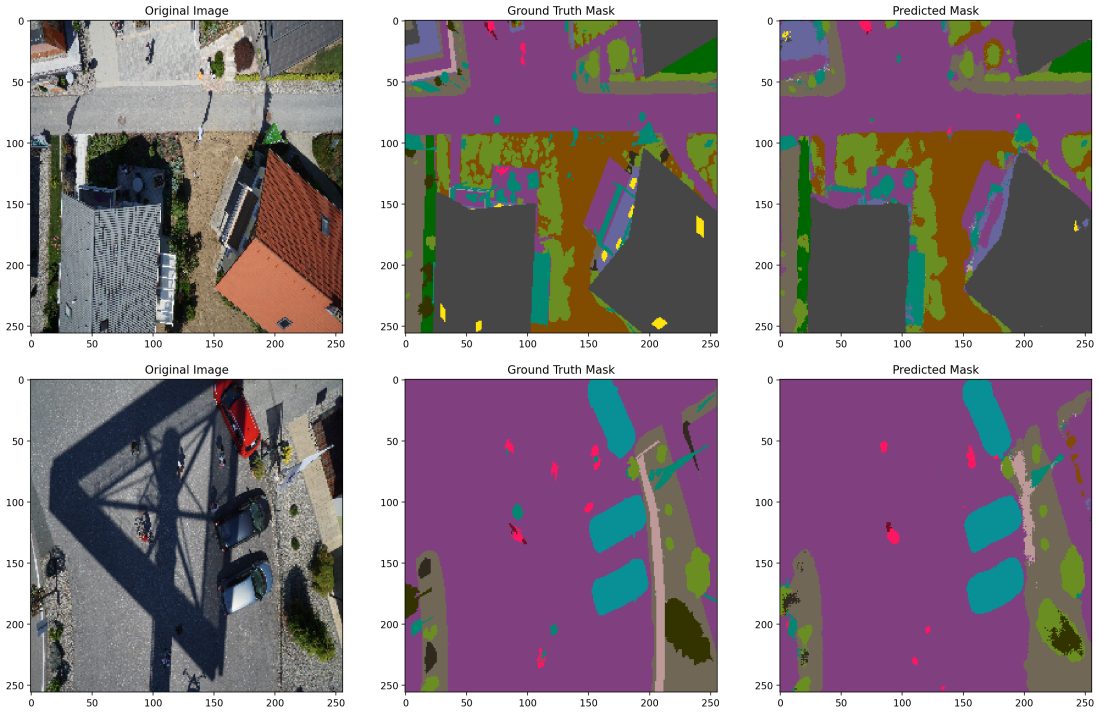

Figure 5: Input images for validation (left column), ground truth (center column), and predicted mask or inference (right column).

The results shown in Table 3 indicate a relatively high average accuracy in semantic segmentation of aerial images using the proposed neural network. The average accuracy across all classes was 0.791, suggesting that the model can correctly identify most objects in the image.

Looking at the results by class, segmentation of objects like pools, persons, dogs, and bicycles had quite high accuracy, surpassing 0.8 in each case. On the other hand, objects like fences, obstacles, and dirt areas had lower accuracy, possibly due to the difficulty of distinguishing these objects from their surroundings.

Table 3
Semantic segmentation results

Class	Jaccard Coefficient
Unlabeled	1.000000
Paved area	0.892092
Dirt	0.563520
Grass	0.900506
Gravel	0.725012
Water	0.913750
Rocks	0.699329
Pool	0.975321
Vegetation	0.670343
Roof	0.870463
Wall	0.582263
Window	0.695331
Door	0.931847
Fence	0.533866
Fence post	0.809327
Person	0.606276
Dog	0.978098
Car	0.940523
Bicycle	0.795602
Tree	0.831066
Bald tree	0.838161
AR marker	0.863405
Obstacle	0.545169
Conflict	1.000000

It is also observed that the VGG16 and UNet-based neural network achieved a high Dice coefficient for classes of persons, cars, and dogs, with values of 0.606, 0.940, and 0.978, respectively. These classes are of vital importance in security monitoring and traffic management in urban areas.

Overall, the obtained results are promising and suggest that deep learning-based semantic segmentation can be a useful tool in applications requiring detailed understanding of aerial images, such as security surveillance, urban planning, and precision agriculture. However, it is essential to note that the model's accuracy can be influenced by various factors, including image quality, scene complexity, and variability in detected objects.

4.3. Inference Time

An evaluation of the model's inference time was conducted using different hardware configurations, including CPU and GPU. The results show that when running on CPU, an inference rate of 2 fps was achieved, which is quite low for practical real-time applications. On the other hand, when evaluating the model using the GPU without optimization, an inference rate of 20 fps was obtained, representing a significant improvement compared to the CPU. However, by

implementing the optimized model on the GPU, an average inference rate of 50 fps was achieved, demonstrating the importance of optimization to enhance model performance. In general, these results suggest that the implementation of deep learning-based models for real-time semantic segmentation of aerial images is feasible using suitable hardware and optimization techniques.

Inference time is a critical factor in real-time applications, such as the monitoring and analysis of aerial images. In our study, the model's performance in terms of processing speed was evaluated using different hardware configurations. When performing inference on a CPU, the model took an average of 0.5 seconds to process each image, resulting in a frames-per-second (FPS) rate of 2. Implementing the model without optimization on the GPU increased the FPS rate to 20. However, with the implementation of optimization techniques, such as operation fusion and precision conversion, an average FPS rate of 50 on the GPU was achieved. This means that the model could process 50 images per second, highlighting the importance of optimization in improving model performance.

5. Conclusions

In summary, this article has proposed a robust solution for semantic segmentation in aerial imagery, leveraging a neural network architecture amalgamating VGG16 and UNet. The Semantic Drone Dataset served as the cornerstone for training, and the integration of data augmentation techniques further amplified model accuracy.

The achieved results are indeed promising, showcasing a Dice coefficient surpassing 50% for the majority of classes. Noteworthy enhancements in model accuracy were realized through the judicious application of data augmentation and hyperparameter optimization.

In terms of inference time, a substantial boost was evident with GPU utilization and model optimization. This translates to a more streamlined application of the model in real-time scenarios.

The demonstrated efficacy of deep learning-based semantic segmentation opens avenues for improved security and efficiency in diverse domains such as smart city planning and monitoring, precision agriculture, environmental surveillance, and other drone-centric applications.

Acknowledgments

The authors extend their gratitude to CONACYT (National Council of Science and Technology, Mexico) for their support in facilitating this research.

References

[1] H. Xiu, P. Vinayaraj, K.-S. Kim, R. Nakamura, W. Yan, 3d semantic segmentation for high-resolution aerial survey derived point clouds using deep learning (demonstration), ACM, 2018, pp. 588–591. URL: https://dl.acm.org/doi/10.1145/3274895.3274950. doi:10.1145/3274895.3274950.

[2] M. S. Alam, J. Oluoch, A survey of safe landing zone detection techniques for autonomous unmanned aerial vehicles (uavs), Expert Systems with Applications

179 (2021) 115091. URL: https://linkinghub.elsevier.com/retrieve/pii/S0957417421005327. doi:10.1016/j.eswa.2021.115091.

[3] J. Kinahan, A. F. Smeaton, Image segmentation to identify safe landing zones for unmanned aerial vehicles (2021). URL: http://arxiv.org/abs/2111.14557.

[4] J. Gonzalez-Trejo, D. Mercado-Ravell, I. Becerra, R. Murrieta-Cid, On the visual-based safe landing of uavs in populated areas: a crucial aspect for urban deployment, IEEE Robotics and Automation Letters 6 (2021) 7901–7908. doi:10.1109/lra.2021.3101861.

[5] J. A. González-Trejo, D. A. Mercado-Ravell, Monitoring social-distance in wide areas during pandemics: a density map and segmentation approach, CoRR (2021). URL: http://arxiv.org/abs/2104.03361v1. arXiv:2104.03361.

[6] A. A. Cabrera-Ponce, L. O. Rojas-Perez, J. A. Carrasco-Ochoa, J. F. Martinez-Trinidad, J. Martinez-Carranza, Gate detection for micro aerial vehicles using a single shot detector, IEEE Latin America Transactions 17 (2019) 2045–2052. URL: https://ieeexplore.ieee.org/document/9011550/. doi:10.1109/TLA.2019.9011550.

[7] R. Girshick, Fast r-cnn (2015) 1440–1448. URL: http://arxiv.org/abs/1504.08083.

[8] M. Fayyaz, M. H. Saffar, M. Sabokrou, M. Fathy, R. Klette, F. Huang, Stfcn: Spatio-temporal fcn for semantic video segmentation (2016). URL: http://arxiv.org/abs/1608.05971.

[9] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation (2015). URL: http://arxiv.org/abs/1505.04597.

[10] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: A deep convolutional encoder-decoder architecture for image segmentation (2015). URL: http://arxiv.org/abs/1511.00561.

[11] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs (2016). URL: http://arxiv.org/abs/1606.00915.

[12] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn (2017). URL: http://arxiv.org/abs/1703.06870.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need (2017). URL: http://arxiv.org/abs/1706.03762.

[14] E. Okafor, R. Smit, L. Schomaker, M. Wiering, Operational data augmentation in classifying single aerial images of animals, IEEE, 2017, pp. 354–360. URL: http://ieeexplore.ieee.org/document/8001185/. doi:10.1109/INISTA.2017.8001185.

[15] X. Yu, Q. Liu, X. Liu, X. Liu, Y. Wang, A physical-based atmospheric correction algorithm of unmanned aerial vehicles images and its utility analysis, International Journal of Remote Sensing 38 (2017) 3101–3112. URL: https://www.tandfonline.com/doi/full/10.1080/01431161.2016.1230291. doi:10.1080/01431161.2016.1230291.

[16] L. T. Thanh, D. N. H. Thanh, An adaptive local thresholding roads segmentation method for satellite aerial images with normalized hsv and lab color models, 2020. URL: http://link.springer.com/10.1007/978-981-15-2780-7_92. doi:10.1007/978-981-15-2780-7_92.

[17] Y. Zhang, L. Fu, Y. Li, Y. Zhang, Hdfnet: Hierarchical dynamic fusion network for change detection in optical aerial images, Remote Sensing 13 (2021) 1440. URL: https://www.mdpi.com/2072-4292/13/8/1440. doi:10.3390/rs13081440.

[18] Y. Li, R. Chen, Y. Zhang, H. Li, A cnn-gcn framework for multi-label aerial image scene classification, IEEE, 2020, pp. 1353–1356. URL: https://ieeexplore.ieee.org/document/9323487/. doi:10.1109/IGARSS39084.2020.9323487.

[19] R. Ratajczak, C. F. Crispim-Junior, E. Faure, B. Fervers, L. Tougne, Automatic land cover reconstruction from historical aerial images: An evaluation of features extraction and classification algorithms, IEEE Transactions on Image Processing 28 (2019) 3357–3371. URL: https://ieeexplore.ieee.org/document/8630683/. doi:10.1109/TIP.2019.2896492.

[20] S. Liu, W. Deng, Very deep convolutional neural network based image classification using small training sample size, IEEE, 2015, pp. 730–734. URL: http://ieeexplore.ieee.org/document/7486599/. doi:10.1109/ACPR.2015.7486599.

[21] T. U. Graz, Semantic drone dataset, 2023. URL: https://www.tugraz.at/index.php?id=22387.