Enhancing Neural Network Models for Precision and Innovation of Autonomous Line Tracking

Abdulrehman Khan
Department of Electrical, Computer, and Biomedical Engineering
Toronto Metropolitan University
Toronto, Canada
a19khan@torontomu.ca

Hamza Iqbal
Department of Electrical, Computer, and Biomedical Engineering
Toronto Metropolitan University
Toronto, Canada
hamza.iqbal@torontomu.ca

Eamon O'Neill
Department of Electrical, Computer, and Biomedical Engineering
Toronto Metropolitan University
Toronto, Canada
eamonn.oneill@torontomu.ca

Yiru Ding
Department of Electrical, Computer, and Biomedical Engineering Toronto Metropolitan University
Toronto, Canada
yiru.ding@torontomu.ca

Cungang (Truman) Yang
Department of Electrical, Computer, and Biomedical Engineering
Toronto Metropolitan University
Toronto, Canada
cungang@ee.torontomu.ca

***Abstract*— This paper explores advancements in autonomous drone navigation through the implementation of deep learning techniques. Specifically, we focus on enhancing real-time line-following capabilities using convolutional neural network (CNN) architectures such as VGG16 and U-Net. By training these models on diverse datasets and integrating them with drone hardware, we aim to achieve accurate path tracking in complex environments. Through extensive experimentation, we demonstrate the effectiveness of our approach in achieving high navigation accuracy and efficiency. Additionally, we discuss encountered challenges and propose future research directions. Overall, our work contributes to advancing autonomous drone technology for diverse real-world applications.**

Keywords*— *DJI Tello drone, convolutional neural networks (CNN), VGG16, U-Net, deep learning, neural network model, image preprocessing, image segmentation, model optimization, data augmentation, custom dataset, TensorFlow

I. Introduction

The focus of autonomous navigation research has been on advancing neural network models to enhance accuracy and lower loss, allowing for more accurate autonomous navigation . This study contributes by leveraging deep learning to improve industrial operations through precise autonomous navigation. Our efforts concentrate on enhancing autonomous line-following drones using CNN architectures, particularly VGG16 and U-Net. We aim to develop highly precise models for real-time path tracking, trained on diverse datasets across various environmental conditions.

The continual evolution and enhancement of neural network models not only increase efficiency but also elevate safety standards across diverse industrial sectors, thus supporting their pivotal role in enhancing decision-making processes. By harnessing the potential of deep learning algorithms and existing literature on neural neural network model research, our goal is to augment the drone model's proficiency in accurately and efficiently detecting and navigating tasks. This research-driven approach seeks to expand the horizons of autonomous drone technology, unlocking opportunities for deployment across sectors such as agriculture and industrial inspections. Furthermore, the meticulous documentation of our research and model development process is aimed at facilitating knowledge dissemination and fostering advancements in this domain. autonomous drones emerge as valuable assets in many industrial environments, facilitating increased efficiency, mitigated risks, and refined decision-making abilities.

The remainder of this paper is organized as follows: In Section II, we review related literature on autonomous drone navigation and deep learning techniques. Section III provides details of our design and implementation, including dataset collection, model training, and hardware integration. Section IV presents the results of our experiments. Section V delves into performance evaluation. In Section VI, we discuss the conclusions drawn from our work and outline future research directions.

II. Literature Review

Using a CNN-based approach for a line-following robot has shown feasibility for deep learning navigation tasks [1]. Models can be trained on images of predefined paths and achieve promising results in real-world scenarios [1]. Additionally, a deep neural network for real-time autonomous indoor navigation showcases the potential of CNNs in various, dynamic environments [2]. By leveraging CNNs, adequate and robust navigation performance even in cluttered indoor spaces can be achieved.

In the realm of semantic segmentation, the deployability of semantic segmentation networks for fluvial navigation was surveyed, highlighting the effectiveness of CNN-based segmentation models in detecting navigable regions in river environments, laying the foundation for autonomous marine navigation systems [3]. Furthermore, exploring the enhancement of path following drones using image-based sensor matrix, emphasizes the role of visual information in improving navigation accuracy [5].

While CNNs have shown promise in autonomous navigation, recent advancements in deep learning architectures have introduced novel approaches for path recognition and obstacle avoidance. The U-Net architecture has gained popularity in image segmentation tasks due to its ability to preserve spatial information and capture fine details, highlighted by its use within medical industry imaging [6]. In addition, a U-Net based, deep neural network used for autonomous UAV navigation in indoor corridor environments, showcases the effectiveness of U-Net-based models in complex navigation scenarios [4].

With further analyzing and researching various additional literatures, we can summarize the following findings, concepts, techniques, and considerations to apply to our model:

A. Data Augmentation

Utilize data augmentation techniques to increase the diversity of our training dataset. This involves techniques such as random rotations, translations, flips, and brightness adjustments. Augmenting our dataset will aid in model robustness regarding variations in lighting conditions, viewpoints, and other environmental factors.

B. Matrix Analysis Techniques

Incorporating comparable matrix analysis techniques into a drone's model navigation system displays potential for immensely enhancing its ability to follow paths. By refining algorithms and utilizing advanced image processing techniques, we anticipate a shift in drone navigation towards being more responsive and adaptive.

C. Contouring

Implementing contouring methods can enhance comprehension of the intricacies involved in identifying and outlining paths. In addition, taking concepts from adaptive contouring algorithms may potentially aid the model and give the drone the capability to move smoothly through changing environments, going beyond traditional constraints.

D. Transfer Learning with VGG16

Leverage transfer learning with pre-trained VGG16 models by fine-tuning the VGG16 model on the specific line-following drone dataset. We could benefit from the feature extraction capabilities of the pre-trained network and adapt it to our target task. This approach can significantly reduce the amount of labeled training data required and expedite the training process.

E. Image Preprocessing

Implement sophisticated image preprocessing techniques to enhance the quality of input images before feeding them into the neural network. This included operations such as noise reduction, histogram equalization, contrast enhancement, edge detection, and any other computer vision concepts. Preprocessing can help improve the network's ability to extract relevant features from input images and ultimately enhance its performance.

F. Real-time Processing

Optimize our neural network models for real-time processing on resource-constrained platforms like drones. This involves considering model compression techniques, such as quantization and pruning, to reduce the model size and computational complexity while maintaining acceptable performance.

G. Visualization Techniques

The use of visualization techniques can aid in understanding deep learning models better. Techniques such as class model visualization can help gain insights into the inner representations learned by the models and diagnose potential problems.

H. Custom Dataset

Value the importance of creating a custom dataset for training deep learning models. Create a diverse and unique dataset specific to the application to improve performance of the various models.

I. Noise Robustness

The generation of additional images with Gaussian white noise to train a model robust to noise can be considered to enhance performance.

J. Semantic Segmentation

U-Net architecture is widely used for semantic segmentation and has shown superior performance in complex environments such as in medical imaging. It leverages its encoder-decoder structure to capture high-level semantic information and generate precise segmentation masks for road navigation.

- *Feature Extraction:* U-Net focuses on extracting feature information from images, which is crucial for accurate segmentation. One could potentially enhance their model by emphasizing feature extraction and image preprocessing in the initial layers of the network.
- *Polygon Fitting:* The use of polygon fitting methods for navigation line extraction such as using polygon fitting algorithms to extract the navigation line from the segmented images captured by the drone's camera is also considerable.

In summary, existing literature demonstrates the potential of deep learning, particularly CNNs and U-Net architectures, in enhancing autonomous drone navigation. By leveraging these techniques, researchers have made significant strides in addressing the challenges of path following and obstacle detection in dynamic environments. In this paper, we build upon these foundations by proposing a CNN-based approach for real-time line following using autonomous drones, with a focus on the VGG16 and U-Net models.

III. Design And Implementation

With regards to the design of our Convolutional Neural Network model, several design approaches were considered, developed, and implemented:

A. Custom CNN

Previously built upon the utilization of TensorFlow, we developed convolutional neural network architectures tailored to the specific requirements of autonomous drone line following. This offered flexibility in design and optimization but required extensive experimentation and tuning for optimal performance.

B. CNN VGG16

In this approach, we leverage the pre-trained VGG16 architecture for feature extraction. Pre-trained on large image datasets, VGG16 is a widely used convolutional neural network. As a baseline, we used the VGG16 model, which has multiple convolutional and pooling layers, on a large dataset and fine-tuned it using transfer learning techniques.. We benefit from VGG16's strong baseline performance and accelerated convergence. However, training the VGG16 model from scratch can require significant computational resources and data.

C. U-Net

We deployed U-Net architecture for semantic segmentation, enabling pixel-wise classification of path segments and obstacles in the drone's environment. The U-Net architecture consists of a contracting path, a bottleneck, and an expansive path, allowing the model to capture fine details and spatial information. U-Net is specifically designed for tasks like image segmentation and has shown superior performance in complex environments such as in medical imaging. The architecture was applied and analyzed within the context of Drone Line Following. Not only did U-Net offer superior performance in complex environments, but required substantially less computational resources and data for training compared to other architectures.

In terms of the implementation workflow, the design and implementation encompass several key components, which includes:

A. Data collection

Utilize a camera attached to the drone (or phone camera) for real-time image capture of the line to be followed and build our dataset. Collect a large dataset of labeled images under various lighting conditions and angles to use for training the neural network models.

B. Data Preprocessing

Preprocess the collected data for our various model implementations by standardizing image sizes, converting to grayscale, normalizing pixel values, and any other additional preprocessing related tasks to enhance image quality to improve model performance.

C. Neural Network Model Building

Implement various models: a built-in custom CNN model, VGG16 CNN architecture model, and a U-Net CNN architecture model. Specified model architectures, including layers, activation functions, and hyperparameters. Utilize TensorFlow framework for model development and optimization.

D. Model Training

Train the models using the labeled dataset, adjusting parameters such as validation split, epochs, and batch size.

E. Model Testing

Test the trained models in real-time using the drone's camera to detect and follow the line. Evaluated model accuracy and performance under different testing conditions.

F. Model Refinement

Identify areas for improving model performance by creating plots, such as reducing latency and loss. Analyze testing results and adjust the model architecture.

IV. Results

Using autonomous drones, we assessed the effectiveness of deep learning models for real-time line following. Using DJI Tello drone with a camera, we navigated predefined paths. A training subset was used to train the model, and a validation subset to evaluate its performance. The model was tested under a variety of lighting conditions.

The performance evaluation encompasses three distinct models: the conventional CNN model, the VGG16 model, and the U-Net model. Each model underwent evaluation utilizing identical parameters, including batch size, number of epochs, and validation split.

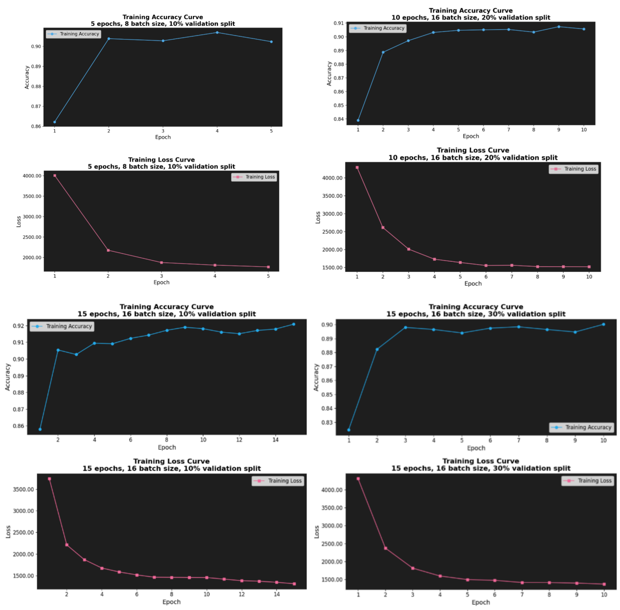

Fig. 1. Performance of conventional CNN model.

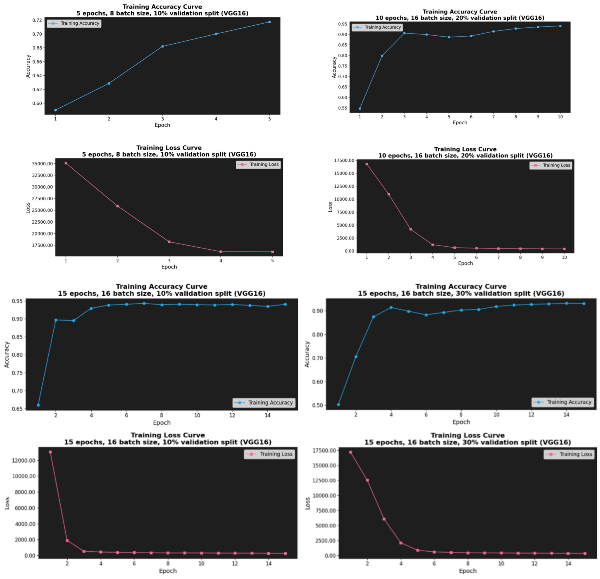

Fig. 2. Performance of the VGG16 model.

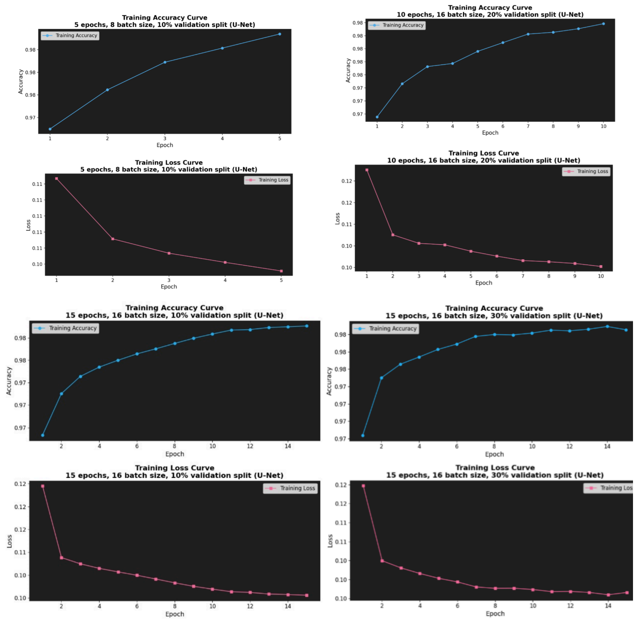

Fig. 3. Performance of the U-Net model.

Through graphical analysis, we identified the most efficient and effective parameters and model by adjusting batch sizes, epochs, and validation splits. Based on the analysis, a batch size of 16, 10 epochs, and a validation split of 20% were found to be optimal for each model.

In general, larger batch sizes often result in faster training times but may lead to less stable convergence. Conversely, smaller batch sizes can produce greater convergence but may take longer to train. A batch size of 16 offered a suitable trade-off between stability and training speed.

The training accuracy of each model increased as the number of epochs increased, up to a certain point where the accuracy plateaued. We found that 10 epochs were optimal, as the models had not yet overfit, and the training accuracy had reached a high level.

The validation split also played a crucial role in the models' performance, with a validation split of 20% outperforming a validation split of 10%. This indicates that the models could accurately predict both new and unseen data, in addition to the training set.

The U-Net model consistently outperformed the conventional CNN and VGG16 models, achieving 98% accuracy and a mere 0.1 loss when trained with the same parameters. This suggests that the U-Net model is more suitable, given its minimal loss and high accuracy.

V. Performance Evaluation

This section predicts the optimal model and parameters through an analysis of the performance results.

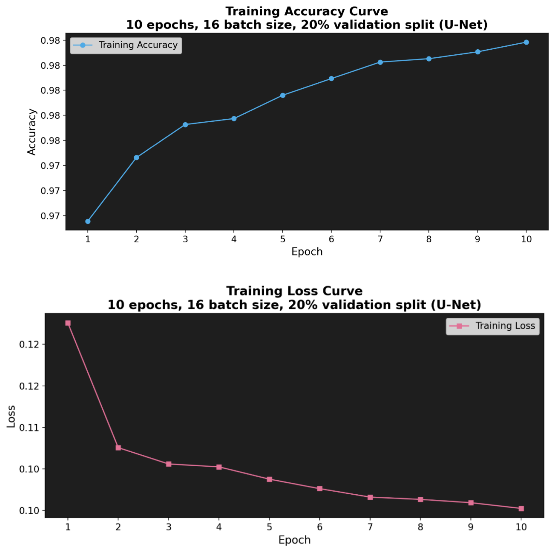

Fig. 4. Plots of the U-Net model with 10 epochs, 16 batch size, 20% validation split.

Fig. 4 illustrates the most efficient and effective model and parameters. Overall, the analysis of the performance graphs indicates that the U-Net model with a batch size of 16, 10 epochs, and a validation split of 20% is optimal for this particular task. However, it is important to note that further research and testing may be required to determine the ideal model and parameters for specific datasets and issues.

- *Batch Size:* The performance results highlight the trade-off between stability and training speed associated with batch sizes. A batch size of 16 is identified as optimal, providing a balance between training speed and convergence stability.
- *Epoch:* It's observed that training accuracy increases with the number of epochs up to a certain point before plateauing. 10 epochs are determined to be optimal, indicating that the model achieves high accuracy without overfitting.
- *Validation Split:* The validation split, which influences the model's ability to generalize to new data, is evaluated. A validation split of 20% is found to be more effective compared to 10%, indicating the model's robustness in predicting unseen data.

- *Model Performance:* Through graph analysis, it's concluded that the U-Net model outperforms the conventional CNN and VGG16 models in terms of accuracy and loss. The U-Net model achieves 98% accuracy and minimal loss, making it more suitable for the project's requirements.

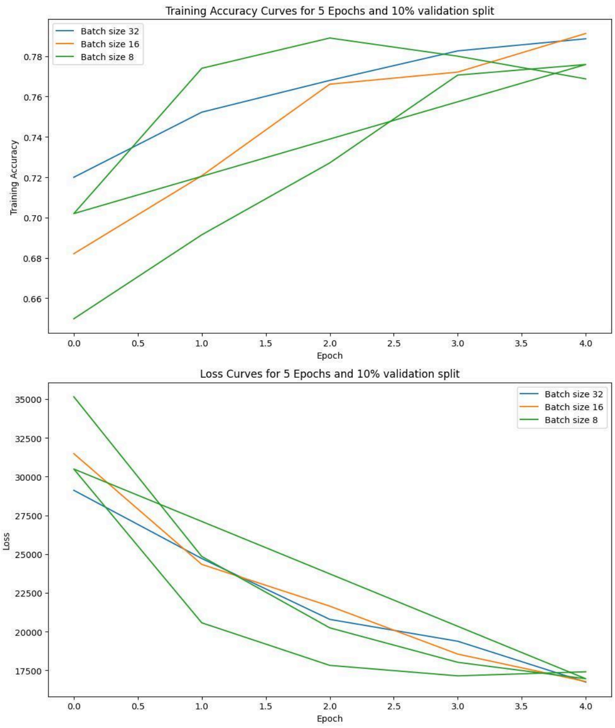

Fig. 5. Plots of the Previous Year's Tensorflow Flow CNN Model with 5 epochs, various batch sizes, 10% validation split.

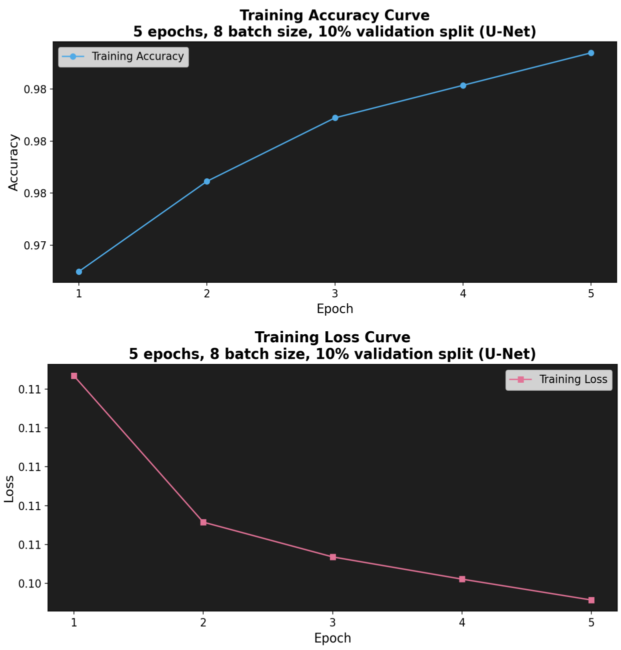

Fig. 6. Plots of the U-Net Model with 5 epochs, 8 batch size, 10% validation split.

In comparison with previous iterations, significant improvements are evident. Previous model achieved approximately 86% accuracy after approximately 10 epochs, with a loss reduced to around 10000. In contrast, the U-Net model in the current implementation achieves over 98% accuracy within 5 to 10 epochs, with a loss of less than 0.1. This remarkable enhancement underscores the efficacy of the U-Net architecture in accurately tracing the designated path while minimizing error, representing a substantial advancement in autonomous line-following capabilities.

Our results indicate significant improvements over conventional CNN architectures, with the U-Net model achieving higher accuracy and robustness in real-world scenarios. Additionally, the performance of the models was consistent across different lighting conditions, demonstrating their reliability for autonomous navigation tasks.

VI. Conclusion

Our research developed and evaluated deep learning models for real-time line following with autonomous drones, showcasing their effectiveness in accurately navigating predefined paths. The U-Net model surpassed traditional CNN architectures like VGG16 and the custom CNN, demonstrating superior accuracy, and precision indicating its real-world viability.

We've established deep learning models more capable of autonomously navigating paths with higher accuracy and minimal loss, laying the groundwork for applications in agriculture, surveillance, search and rescue, and other relevant fields. Insights into model design, implementation, and performance metrics were provided. Despite progress, future research aims to extend experiments to diverse environments and integrate reinforcement learning for adaptive decision-making. Collaboration with industry partners and addressing limitations like controlled environments are key priorities.

- *Implications:* The successful deployment of deep learning models for autonomous drone navigation has significant implications for various industries and domains. These models can enhance efficiency, accuracy, and safety. By leveraging advanced AI techniques, we can unlock new opportunities for innovation and progress in aerial robotics.
- *Future Directions:* Despite the promising results, there are several avenues for future research and development. Firstly, we aim to extend our experiments to more diverse and challenging environments, including outdoor terrains and urban landscapes. Additionally, we plan to explore the integration of reinforcement learning techniques to enable adaptive and intelligent decision-making by autonomous drones.

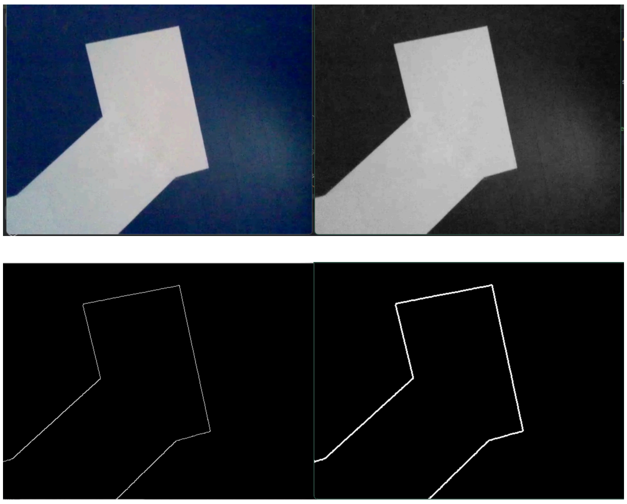

Fig. 7. Image Transformation Techniques.

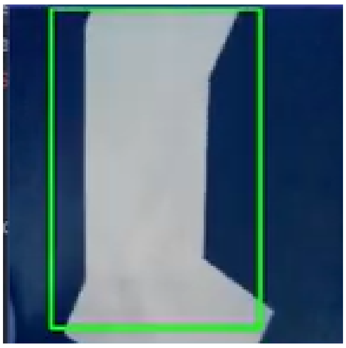

Fig. 8. VGG16 Model Line Tracking With Bounding Boxes.

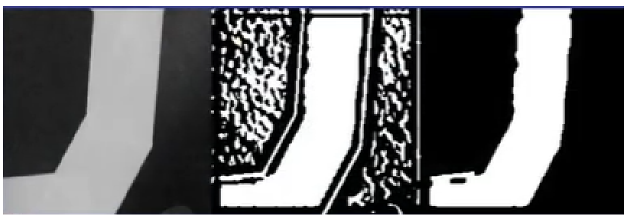

Fig. 9. U-Net Image Segmentation Process.

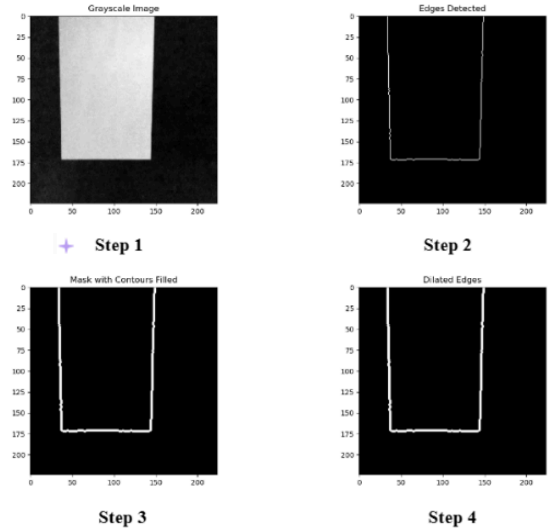

Fig. 10. U-NET Image Labeling Process.

References

[1] Ahmad, N. (2019, April 3). Line Follower Robot using CNN. Towards Data Science. Retrieved from https://towardsdatascience.com/line-follower-robot-using-cnn-3fe73165ea54

[2] Kim, D. K., & Chen, T. (2015). Deep neural network for real-time autonomous indoor navigation. arXiv preprint arXiv:1511.04668.

[3] Lambert, R., Li, J., Chavez-Galaviz, J., & Mahmoudian, N. (2023). A Survey on the Deployability of Semantic Segmentation Networks for Fluvial Navigation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 255-264).

[4] Padhy, R. P., Verma, S., Ahmad, S., Choudhury, S. K., & Sa, P. K. (2018). Deep neural network for autonomous UAV navigation in indoor corridor environments. Procedia computer science, 133, 643-650.

[5] Yarru, L. S. K., & Penugonda, T. F. (2023). Enhancing path following drone using image-based sensor matrix (Bachelor of Science in Electrical Engineering). Blekinge Institute of Technology, Faculty of Engineering, Karlskrona, Sweden.

[6] Yu, J., Zhang, J., Shu, A., Chen, Y., Chen, J., Yang, Y., ... & Zhang, Y. (2023). Study of convolutional neural network-based semantic segmentation methods on edge intelligence devices for field agricultural robot navigation line extraction. Computers and Electronics in Agriculture, 209, 107811.