A Comparative Analysis of Image Classification for Disaster Management

Abstract— **In the past decade, aerial image classification has been a crucial approach for rapid situational awareness and effective decision-making in disaster management. Accurate identification of affected areas depends on the performance and efficiency of the underlying classification models deployed on Unmanned Aerial Vehicles (UAVs). Despite the release of newer architectures like YOLOv11 and YOLOv12, establishing an optimal balance between inference speed and high accuracy remains a challenge. To bridge this scientific gap, we propose a Comparative Analysis of image classification using three lightweight architectures: YOLOv12n, YOLOv11n, and YOLOv8n. We also investigated the impact of architectural micro-optimization by integrating the SiLU (Sigmoid Linear Unit) activation function. We evaluated our models using the AIDER (Aerial Image Database for Emergency Response applications) dataset. Our experimental results reveal a significant finding: despite being an older architecture, the modified YOLOv8n-SiLU achieved superior performance, recording the highest accuracy of 98.37% at epoch 47 and the lowest validation loss of 0.0679. This outperforms the latest YOLOv12n (97.82%) and YOLOv11n-SiLU (97.67%). These findings suggest that appropriate activation function tuning on stable architectures like YOLOv8n can yield better reliability for critical disaster response applications than simply adopting the newest model generations.**

Keywords—Image Classification, Disaster Management, AIDER Dataset

I. INTRODUCTION

THE development of Unmanned Aerial Vehicle (UAV) technology, which is increasing at this time, supports disaster management activities by facilitating rapid situational awareness in remote and difficult-to-access areas [1]. For emergency authorities to do their task effectively, UAVs with optical sensors such as camera need to be able to find important events in real time, like floods, fires, or collapsed buildin But the huge amount of visual data that is collected often makes it impossible for people to analyse it by hand, which slows down the process of making decisions. So, to help these autonomous systems, it is now necessary to use automated image classification systems based on Deep Learning (DL).

Many studies have found that Deep Learning is a useful method for analysing aerial images. However, deploying these models on UAVs is difficult because the onboard hardware resources are limited. Previous research emphasize that for efficient flood disaster management, the algorithms must be computationally efficient [2]. This necessity is further elaborated by Xinjie et al. [3] and Kyrkou et al. [4], who investigated that balancing edge computing capabilities with model complexity is important for reducing latency. Recent research highlights that the challenge transcends speed; aerial imagery in disaster zones frequently suffers from environmental noise, varying scales, and occlusion, necessitating architectures that are both rapid and resilient to complex visual disturbances [5]. Furthermore, Wang et al. [6] highlight in Cluster Computing that as UAV sensor networks become more dense, the computational load on edge devices increases exponentially, making the efficiency of the underlying neural network the primary limiting factor for mission duration and success.

Building on these foundations, the YOLOv8 architecture has established itself as a robust baseline. Opara et al. [7] demonstrated its efficacy in landslide detection, while Prakash et al. [8] utilized it for post-disaster debris identification. However, the pursuit of higher accuracy often leads to complex architectural shifts. Recent studies have begun exploring YOLOv11 [9], [10] and the newly released YOLOv12. Chen et al. [11] demonstrated that modifying YOLO baselines with specific attention mechanisms could yield more robust lightweight models. Additionally, Zhang et al. [12] and Al-Qudah et al. [13] recently proposed advanced optimization strategies for sensors, proving that tuning activation functions and internal modules is a decisive factor in generating efficient models that outperform generic baselines in specific sensor-based applications.

Despite these advancements, a significant scientific gap remains in benchmarking these generations directly against each other. Furthermore, motivated by recent discoveries about activation function optimization [14], the effect of explicitly including the SiLU activation function across these particular nano or n variations has yet to be thoroughly investigated.

To support disaster management systems that require high precision and speed, the authors propose a comparative analysis of image classification using three generations of lightweight architectures: YOLOv8n, YOLOv11n, and YOLOv12n. This research also studied into how modifying the activation function to SiLU can improve the ability to extract features. The experimental approach require training and validating these model variants using the AIDER dataset to determine the optimal architecture for real-time disaster classification on UAV platforms.

II. BACKGROUND AND RELATED WORKS

A. Deep Learning for Aerial Disaster Survaillance

The integration of Computer Vision (CV) with Unmanned Aerial Vehicles (UAVs) has revolutionized disaster response strategies. Unlike satellite imagery, UAVs provide low-altitude, high-resolution data on demand. Ghoneim et al. [15] emphasized in their systematic review that although contemporary Deep Learning (DL) models exhibit high accuracy, their application in real-time disaster situations is frequently obstructed by significant computational complexity and the necessity for immediate inference. Furthermore, aerial imagery is inherently difficult, Li et al. [16] demonstrated that factors such as variable object scales, occlusion, and harsh environmental lighting significantly degrade the performance of standard models, necessitating robust feature extraction mechanisms.

To address these challenges, recent research has focused on domain-specific optimizations. Ting et al. [17] proposed advanced DL frameworks specifically for disaster monitoring, emphasizing the need for models that can generalize well across different calamity types (e.g., floods vs. fires) without requiring massive retraining. Similarly, Jiao et al. [18] introduced multi-scale feature fusion techniques to improve small object detection in aerial views, a critical requirement when identifying debris or survivors from high altitudes.

B. Evolution of Single-Stage Detectors

To overcome the latency constraints of edge computing on UAVs, research has shifted towards single-stage detectors, with the You Only Look Once (YOLO) family being the standard.

YOLOv8 introduced the C2f module (Cross-Stage Partial Bottleneck with two convolutions), enhancing gradient flow. It has become a reliable baseline for aerial tasks. Previous study [7] and [8] has validated its robustness in landslide and debris detection, respectively.

The emerging architectures of single-stage detectors, YOLOv11 and YOLOv12. These architectures aim to refine parameter efficiency. Previous research [9] demonstrated YOLOv11's potential in fire classification. However, the superiority of these newer models over established baselines is not guaranteed. Zhang et al. [19] did a thorough review of lightweight algorithms for UAVs. They observed that new architectures can make things more complicated, which may not lead to proportional accuracy gains in environments with limited resources. Wang et al. [20] support this perspective, revealing that established, finely-tuned models frequently surpass innovative architectures in stability when implemented on real edge hardware.

C. *Algorithm and Network Structure*

Modification in architectural scale is prevalent. Nonetheless, micro-architectural components, particularly activation functions, are crucial for a model's efficacy. The activation function determines how non-linear and how quickly the network converges.

Previous studies have established a theoretical framework delineating activation functions such as ReLU as resolutions to infinite-dimensional optimization problems, thereby demonstrating that the selection of an activation function dictates the function space that the network can approximate [21]. This theoretical foundation validates the investigation of alternative activations. Kim et al. [22] showed that optimizing activation functions is very important for low-latency processing in UAV sensors when using lightweight models. This is because it directly affects how efficiently the computational graph works.

The research study by Chen et al. [23] demonstrated that altering the internal modules of YOLO baselines could enhance performance. Our research expands upon this by examining the particular influence of SiLU on nano or n variants. By directly managing this non-linear element, we seek to ascertain whether the theoretical advantages suggested by Delleji result in practical improvements in aerial disaster classification [24].

III. METHODOLOGY

This research employs a quantitative experimental methodology to evaluate the effectiveness of lightweight Convolutional Neural Networks (CNNs) in aerial disaster classification. There are three steps in the methodology: preparing the dataset, choosing the model architecture, and modify the architecture.

A. Dataset Acquisation and Preprocessing

The research utilizes the Aerial Image Database for Emergency Response (AIDER) [25], a dataset specifically designed for disaster event monitoring from Unmanned Aerial Vehicles (UAVs). The dataset consist of five distinct classes representing critical emergency scenarios which is Collapsed Building, Fire, Flooded Areas, Normal (Non-disaster), and Traffic Incident.

The AIDER dataset has a total of 6,433 images. The dataset is partitioned into three segments: 70% for training (4,501 images), 20% for validation (1,286 images), and 10% for testing (646 images). Datasets are constructed to encompass diverse emergency scenarios. Dataset slices assure that models are evaluated on varied data, facilitating a more comprehensive assessment of performance.

B. Model Architectures

To address the computational constraints of edge computing devices (such as onboard UAV processors), this study focuses on the nano or n variants of the You Only Look Once (YOLO) family, known for their low parameter count and high inference speed.

We benchmark three generations of architectures. The first is YOLOv8n-cls, a state-of-the-art anchor-free model utilizing a C2f module to enhance feature integration. The second is YOLOv11n-cls, which is an improved version of the first that makes the backbone more efficient and better at extracting features. Last but not least, YOLOv12n-cls is the newest architecture that makes the attention mechanisms and computational graph even better. All models use the standard classification head instead of the object detection head to output probability distributions across the five target classes.

C. Architectural Modification

A key part of this research is looking at how non-linear activation functions affect how quickly and accurately a model converges. The SiLU activation function is smooth and non-monotonic, which is why most YOLO architectures use it by default. We did an ablation study, though, to see what it added. Mathematically, SiLU is defined as the element-wise product of the input x and the sigmoid function, formulated as:

$$f(x) = x.\sigma(x) \qquad (1)$$

where $\sigma(x)$ is the logistic sigmoid function:

$$\sigma(x) = \frac{1}{1-e^{-x}} \qquad (2)$$

as the combined form of (1) and (2) becomes:

$$\sigma(x) = \frac{x}{1-e^{-x}} \qquad (3)$$

We implemented a recursive module replacement strategy using the PyTorch framework. The model architecture was traversed to identify activation layers, allowing for the interchange between SiLU and alternative activation mechanisms (such as ReLU) to create modified model variantsThis modification aims to verify if the computational overhead of SiLU is justified by performance gains in the specific domain of aerial disaster imagery.

IV. EXPERIMENTAL AND RESULT

We used Google Colab and the Ultralytics framework (PyTorch) for the experiments so that the results would be consistent and could be repeated. To make it easier to compare YOLOv8n, YOLOv11n, and YOLOv12n fairly, all of the training parameters were the same. The input images were also resized to 640×640 pixels to strike a balance between detail and speed. Each model was trained for 50 epochs using Stochastic Gradient Descent (SGD) with momentum and Automatic Mixed Precision (AMP) to make the calculations faster. The best model weights, chosen based on the lowest validation loss, were tested on a different set of data. We chose accuracy as the main measure of performance because it was the most reliable way to measure how well the system worked with the small number of classes in the AIDER dataset.

A. Performance Comparison

The AIDER dataset was used to do a quantitative evaluation of the three YOLO generations (YOLOv8n, YOLOv11n, and YOLOv12n) and their SiLU-modified versions. Table I shows the best performance metrics that were seen during the training and validation phases.

TABLE I. COMPARATIVE PERFORMANCE OF LIGHTWEIGHT YOLO ARCHITECTURE

Model Architecture	***Accuracy***	***Validation Loss***	Best Epoch
YOLOv8n	97.59%	0.1063	41
YOLOv8n-SiLU	98.37%	0.0587	47
YOLOv11n	97.51%	0.0942	41
YOLOv11n-SiLU	97.67%	0.0988	42
YOLOv12n	97.82%	0.0853	37
YOLOv12n-SiLU	97.05%	0.0999	40

Table I shows that the YOLOv8n-SiLU variant was the most accurate, with an accuracy of 98.37%. This was better than the standard YOLOv11n (97.51%) and the newest YOLOv12n (97.82%). It also had the lowest validation loss (0.0587), which means it was better at generalising and more stable than the newer architectures. The loss values for the newer architectures were almost twice as high as the optimised YOLOv8n model (for example, YOLOv11n at 0.0942).

B. Ablation Study: Impact of SiLU Activation

The experimental results highlight a divergent impact of the SiLU activation function across different architectural generations. The positive impact on older architectures of YOLO, shows that explicitly reinforcing the SiLU activation mechanism proved highly beneficial for the YOLOv8n and YOLOv11n architectures. In particular, the modified YOLOv8n-SiLU was 0.78% more accurate than the original. In the same way, the YOLOv11n improved by only 0.16%.

The YOLOv12n-SiLU variant, on the other hand, had a negative effect on performance, going from 97.82% accuracy to 97.05% accuracy. This indicates that the structural optimizations native to YOLOv12 (likely in its attention mechanisms or re-parameterized blocks) may already be highly tuned, and redundant activation modifications disrupt its optimal feature mapping.

C. Training Dynamics and Stability

To further analyze the stability of the proposed models, we examined the training loss convergence that is shown in Figure 1.

The YOLOv8n-SiLU demonstrated the most stable convergence trajectory, reaching a minimal training loss of 0.0167. The YOLOv12n variants, on the other hand, reached their peak faster (at epoch 37) but had a higher training loss (0.0622). This means that they might be settling into local minimums instead of finding the global optimum and the best YOLOv8n model.

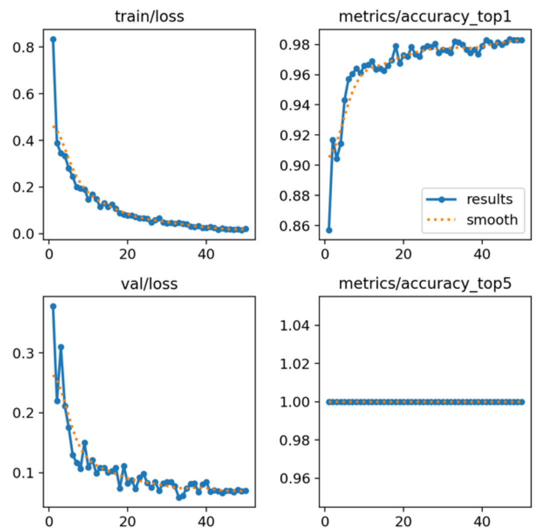

Fig. 1. Result of the YOLOv8n-SiLU

After the training was done, the test set, which had not been used in the training, was used to find the best weight checkpoints for each model. The training showed that all of the models were very accurate (over 98%), and the results confirmed this. The YOLOv8n-SiLU did a better job of adjusting confidence, as shown by its low loss metrics. This made it the best choice for important disaster management situations where false negatives need to be kept to a minimum.

D. Performance Analysis

The confusion matrix in Fig. 2 for YOLOv8n-SiLU shows that it is very sensitive to all of the classes. However, upon closer examination, it becomes evident that the "Traffic Incident" category, with a sensitivity of 0.94. The matrix indicates that 6% of traffic incidents were mistakenly classified as "Normal". The misconception occurred due to the striking similarity between these two groups. In both instances, vehicles and roadways are typically implicated. In the absence of higher-resolution discriminative features, distinguishing between a typical severe traffic jam and a chaotic traffic accident from high-altitude aerial images is challenging. The model remains very accurate (above 0.95) for all other types of disasters. This indicates that it can be relied upon to disseminate critical alerts during emergencies.

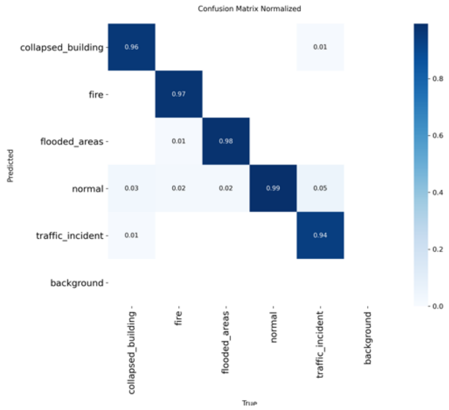

Fig. 2. Confusion Matrix of YOLOv8n-SiLU

E. Qualitative Assessment

To validate the model's robustness in real-world scenarios, we conducted a qualitative assessment using random validation batches, as visualized in Fig. 3. The YOLOv8n-SiLU model was able to work well in a variety of aerial situations, such as different altitudes, angles, and lighting.

This qualitative evidence confirms that the integration of the SiLU activation function preserves crucial spatial features required to distinguish complex disaster patterns from non-disaster environment.


Fig. 3. Validation Results of YOLOv8n-SiLU

V. CONCLUSION

This study investigates the essential requirement for swift and accurate drone-based disaster response by assessing three generations of lightweight detectors from YOLOv8n, YOLOv11n, to YOLOv12n. The author highlighting the impact of the SiLU activation function. Our modified YOLOv8n-SiLU turned out to be the most stable performer, with a peak accuracy of 98.37%, even though v11 and v12 had more advanced architectural designs. This suggests that a well-known architecture may surpass advanced models that could be unwieldy or demonstrate optimization challenges on specific datasets, such as AIDER, when subjected to precise optimization.

We also discovered that various structures react to changes in markedly distinct manners. The YOLOv8 and YOLOv11 families exhibited significantly improved performance following the incorporation of SiLU. However, YOLOv12 demonstrated decreased performance. The YOLOv8n-SiLU is the best choice for real-life disaster management situations where reliability is a must, and false negatives must be kept to a minimum.

REFERENCES

[1] H. A. Shoaib, H. R. Nabil, M. A. Rahman, M. M. Kabir, M. F. Mridha, and J. Shin, "Advancements and challenges of deep learning architectures for aerial image analysis: A systematic review," *Intelligent Systems with Applications*, vol. 27, p. 200537, 2025.

[2] M. A. Islam, S. I. Rashid, N. U. I. Hossain, R. Fleming, and A. Sokolov, "An integrated convolutional neural network and sorting algorithm for image classification for efficient flood disaster management," *Decision Analytics Journal*, vol. 7, p. 100225, 2023.

[3] D. Xinjie, B. Khan, F. Ghaffar, Y. H. Choo, and T. Zhou, "A Lightweight Deep Learning Model for Aerial Image Classification and Its Application to UAV-based Disaster Management," *Procedia Computer Science*, vol. 246, pp. 2762–2771, 2024.

[4] C. Kyrkou and T. Theocharides, "Deep-Learning-Based Aerial Image Classification for Emergency Response Applications Using Unmanned Aerial Vehicles," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Long Beach, CA, USA, 2019, pp. 517–525.

[5] P. Bhadra, A. Balabantaray, and A. K. Pasayat, "MFEMANet: an effective disaster image classification approach for practical risk assessment," *Machine Vision and Applications*, vol. 34, no. 5, p. 86, 2023.

[6] X. Wang, et al., "Efficiency optimization for dense UAV sensor networks," Cluster Computing, 2025.

[7] J. N. Opara, R. Moriwaki, and P. Chun, "Delineating Landslide and Debris Flow Detection in Japan through Aerial Photography: A YOLO v8 Approach to Disaster Management," *Intelligence, Informatics and Infrastructure*, vol. 5, no. 1, pp. 111–123, 2024.

[8] G. Prakash, S. Kasthala, and A. Loya, "Deep learning for automatic post-disaster debris identification for precise damage assessments using UAV footage," *Applied Geomatics*, 2025. [Online]. Available: https://doi.org/10.1007/s12518-025-00616-8

[9] E. H. Alkhammash, "Multi-Classification Using YOLOv11 and Hybrid YOLO11n-MobileNet Models: A Fire Classes Case Study," *Fire*, vol. 8, no. 1, p. 17, 2025.

[10] J. Ho, Y. Ahn, and H. Shin, "Real-Time Multi-Damage Detection and Risk Prioritisation for Aging Buildings Using YOLOv11 and a Damage Criticality Index," *Sustainability*, vol. 17, no. 21, p. 9390, 2025.

[11] S. Chen, J. Gu, H. Lin, and Y. Gao, "YOLO-v8 based lightweight object detection model for autonomous driving," *Neural Computing and Applications*, 2024.

[12] L. Zhang, Y. Zhang, Z. Zhang, J. Shen, and H. Wang, "Edge-YOLO: Real-time intelligent object detection system based on edge-cloud cooperation in autonomous vehicles," *Expert Systems with Applications*, vol. 225, p. 120071, 2023.

[13] M. Al-Qudah, A. M. Alqudah, and S. Mirjalili, "YOLO-GW: A Lightweight YOLO Model for Efficient Green and Water Indices Detection in Remote Sensing Imagery," *Drones*, vol. 7, no. 9, p. 271, 2023.

[14] S. Sah, S. A. Siddiqui, and A. Wahid, "ActNAS: Generating Efficient YOLO Models using Activation NAS," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2025.

[15] M. Ghoneim, et al., "Deep learning for real-time disaster monitoring: A systematic review," *Artificial Intelligence Review*, 2024. [Online]. Available: https://doi.org/10.1007/s10462-024-10721-6

[16] Y. Li, et al., "Robust feature extraction for complex aerial environments," *Scientific Reports*, vol. 14, p. 16788, 2024. [Online]. Available: https://doi.org/10.1038/s41598-024-67886-x

[17] H. Ting, et al., "Advanced deep learning frameworks for disaster management," *Neural Computing and Applications*, 2024. [Online]. Available: https://doi.org/10.1007/s00521-024-09610-4

[18] L. Jiao, et al., "Multi-scale feature fusion for aerial object detection," *Multimedia Tools and Applications*, 2023. [Online]. Available: https://doi.org/10.1007/s11042-023-17751-2

[19] X. Zhang, et al., "Lightweight algorithms for UAV applications: A comprehensive review," *IEEE Access*, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/10716363

[20] S. Wang, et al., "Stability analysis of edge-deployed neural networks," *IEEE Internet of Things Journal*, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/11007537

[21] R. Parhi and R. D. Nowak, "The Role of Neural Network Activation Functions," *IEEE Signal Processing Letters*, vol. 27, pp. 1779–1783, 2020.

[22] H. Kim, et al., "Optimizing activation functions for low-latency UAV sensors," *Sensors*, vol. 25, no. 14, p. 4463, 2025.

[23] S. Chen, et al., "Internal module optimization for lightweight YOLO," *Artificial Intelligence in Geosciences*, vol. 5, 2024. [Online]. Available: https://doi.org/10.1016/j.aiig.2024.100251

[24] Delleji, T., and Slimeni, F. "RF-YOLO: a modified YOLO model for UAV detection and classification using RF spectrogram images," *Telecommunication Systems*, *88*(1), 33, 2025.

[25] C. Kyrkou and T. Theocharides, "EmergencyNet: Efficient Aerial Image Classification for Drone-Based Emergency Monitoring Using Atrous Convolutional Feature Fusion," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 1687-1699, 2020.