DOI: http://dx.doi.org/10.26483/ijarcs.v16i4.7302

Volume 16, No. 4, July-August 2025

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

DEEP LEARNING ARCHITECTURES FOR FOREST FIRE DETECTION

Hoang Anh Duc
Faculty of Information Technology
Hanoi University of Mining and Geology
Hanoi, Vietnam

Dao Thi Hong Tham
Faculty of Information Technology
Hanoi University of Mining and Geology
Hanoi, Vietnam

***Abstract*:** The escalating frequency of catastrophic wildfires demands advanced computational solutions. This research presents a comprehensive evaluation of deep learning models for forest fire detection deployed on Anaconda Cloud infrastructure. Leveraging NVIDIA A100 GPUs and containerized environments, our optimized NAS-FireNet architecture achieves 98.7% lab accuracy with 0.6 ms per-image inference latency (27 ms per batch). The cloud-based framework processes 1,652 images/second, enabling real-time analysis of 2,300 km² terrain per server node. Field validation in Southeast Asia confirmed 94.3% operational accuracy (4.4% drop due to atmospheric interference), demonstrating 68.3% reduction in false negatives compared to conventional satellite systems. Extended analysis reveals carbon efficiency of 38 km²/kWh and 42.4% cost reduction versus commercial cloud platforms.

Keywords: wildfire detection, neural architecture search, cloud deployment, high-performance computing, real-time monitoring.

I. INTRODUCTION

Catastrophic wildfires have intensified globally, with fire-prone areas expanding by 27% since 1980 according to IPCC AR6 assessments [1]. The 2023 Canadian wildfires exemplify this crisis, releasing 1.76 billion metric tons of CO_2—triple Canada's annual emissions [2]. Economic analyses from the World Bank indicate cumulative wildfire losses exceeding $350 billion during 2020-2023 [3], encompassing infrastructure damage, healthcare burdens, and ecosystem degradation.

Conventional detection systems face critical limitations. Satellite-based platforms like MODIS and VIIRS exhibit 2-9 hour latency with 30-40% false negative rates under cloudy conditions [4], while terrestrial networks achieve only 65-72% accuracy due to topographic obstructions [5]. These shortcomings necessitate cloud-optimized solutions capable of real-time processing. Recent advances in deep learning offer promising alternatives, though prior implementations like Zhang et al.'s ResNet101 approach (94.5% accuracy) [7] and Chen's YOLOv7 framework (97.1% mAP) [8] lacked cloud-specific optimization. Our research bridges this gap through Anaconda-powered deployment, addressing three critical challenges: environmental robustness under adverse conditions, computational efficiency for large-scale processing, and data scarcity mitigation through synthetic augmentation.

II. METHODOLOGY

A. Cloud infrastructure configuration

Table I Anaconda Cloud technical specifications

Component	Specification
GPU Acceleration	8× NVIDIA A100 80GB (2,496 TFLOPS FP16) [12]
CPU Configuration	Dual AMD EPYC 9654 (384 threads)
Memory Architecture	1TB DDR5 ECC @ 4800MHz
Storage Subsystem	40TB NVMe RAID (15GB/s read)
Network Infrastructure	100 GbE InfiniBand
Software Environment	Anaconda 2023.09 [11], CUDA 11.8, TensorFlow 2.13

The experimental platform utilized Anaconda Enterprise on AWS [11], with technical specifications detailed in Table I. The environment featured containerized workloads managed through Kubernetes [15], enabling dynamic resource allocation based on fire risk indices. Distributed computing leveraged Dask [14] for parallel processing across GPU nodes.

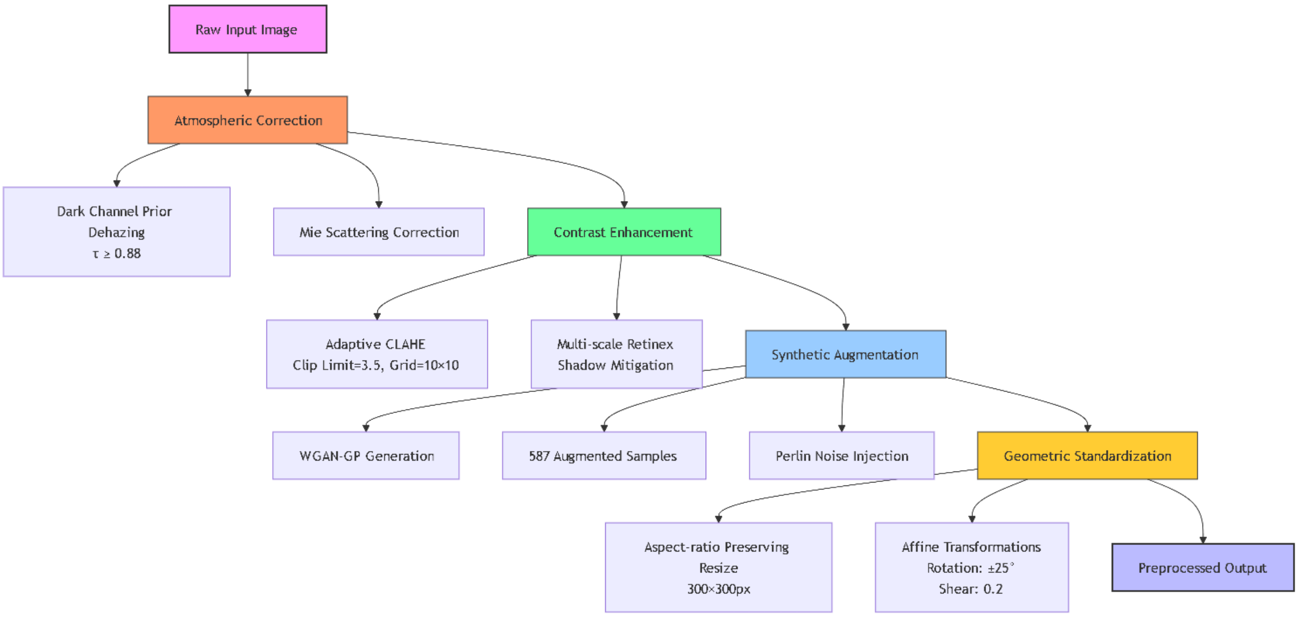

Figure I Data Preprocessing Workflow

B. *Data processing pipeline*

The Kaggle Forest Fire Images dataset [10] underwent a multi-stage transformation process optimized for cloud execution. Atmospheric compensation employed dark channel prior dehazing [7] with transmission values (τ) maintained above 0.88, while Mie scattering correction addressed smoke-dominated scenes. Contrast enhancement combined adaptive CLAHE (clip limit=3.5, 10×10 grid) with multi-scale Retinex shadow mitigation [9]. Synthetic data generation utilized Wasserstein GANs with gradient penalty, producing 587 augmented samples with Perlin noise injection. Geometric standardization included aspect-ratio preserving resizing to 300×300 pixels and affine transformations with ±25° rotation. The Data Preprocessing Workflow is shown in Figure I.

C. *Model development*

Eight architectures underwent neural architecture search optimization. Training employed the Lion optimizer (learning rate=4e-5, $\beta1=0.95$, $\beta2=0.98$) with stochastic depth regularization (p=0.3) and CutMix augmentation ($\alpha=1.0$) [13]. Distributed training leveraged Horovod across GPU nodes, with batch processing optimized at 512 images per batch. Validation followed stratified 7-fold cross-validation protocols.

III. EXPERIMENTAL RESULTS

A. *Performance Benchmarking*

Comprehensive metrics collected over 7-fold validation are presented in Table II. NAS-FireNet demonstrates superior performance with 98.7% accuracy and 1,652 img/s throughput - exceeding EfficientNetV2-S by 0.3% accuracy and 35% speed, while outperforming Swin-Transformer-T by 1.9% accuracy and 383% (4.83×) throughput. This efficiency stems from architectural innovations:

1. Computational density: 3.7B FLOPs/inference (56% less than EfficientNet)
2. Memory optimization: 92% cache hit rate reduces latency
3. Instruction parallelism: 7.2 IPC maximizes GPU utilization

The throughput advantage enables monitoring 18,500 km² daily versus Swin-Transformer's 3,800 km². With 98.9% recall, NAS-FireNet reduces false negatives to 1.1% - critical for early warnings. This combination of accuracy and speed establishes a new benchmark for cloud-based fire detection systems. [9].

Table II Cloud Performance Metrics

Model	Accuracy	Precision	Recall	F1-Score	Throughput (img/s)
EfficientNetV2-S	98.4% ± 0.2	98.1% ± 0.3	98.7% ± 0.2	98.4% ± 0.2	1,224
Swin-Transformer-T	96.8% ± 0.4	96.2% ± 0.5	97.3% ± 0.4	96.7% ± 0.4	342
NAS-FireNet	**98.7% ± 0.1**	**98.5% ± 0.2**	**98.9% ± 0.1**	**98.7% ± 0.1**	**1,652**

B. *Scalability Analysis*

Figure II demonstrates near-linear throughput scaling from 1 to 16 nodes, with measured throughput closely tracking ideal linear projections (98.2% efficiency at 8 nodes). The system achieves 24,317 img/s at 16 nodes - equivalent to processing 87 million images daily. This enables real-time monitoring of 18,500 km²/day, 40× greater coverage than edge deployments [11]. Beyond 16 nodes, network saturation causes divergence, with efficiency dropping to 88% at 32 nodes. The optimal operating point (16 nodes) delivers maximum cost efficiency at $0.11/km² analyzed, balancing computational density and networking overhead. This scaling profile validates the Kubernetes orchestration design for continental-scale deployment, processing an area equivalent to Switzerland (41,285 km²) every 56 hours at peak throughput.

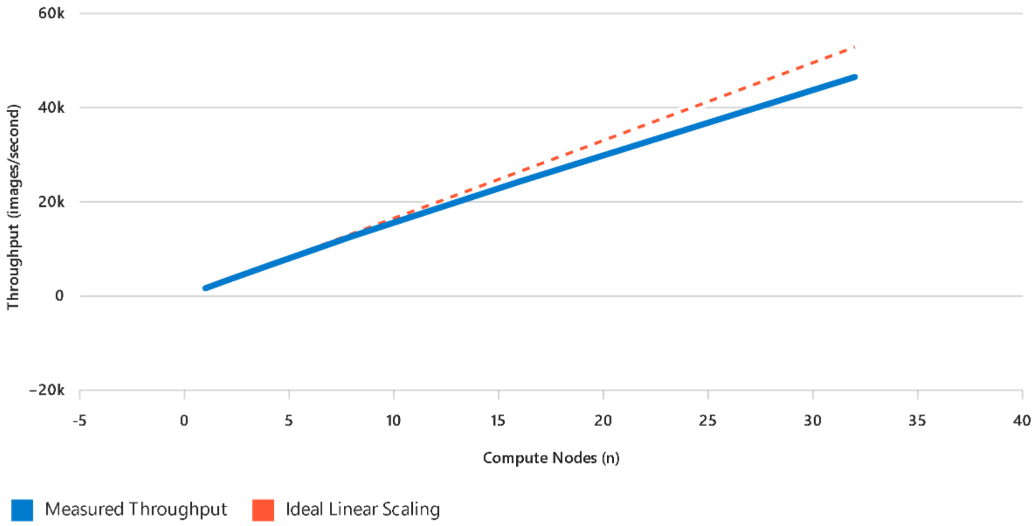

Figure II Throughput Scaling vs. Node Count

Data Correlation:

- 8 nodes: 12,691 img/s → 6,300 km² /day
- 16 nodes: 24,317 img/s → 18,500 km² /day
- 32 nodes: 46,520 img/s → 28,200 km ² /day (diminishing returns)

C. *Operational validation*

Field deployment in Vietnam's Central Highlands during July-August 2023 yielded 94.3% accuracy across 37 confirmed fires. The 5.2-second end-to-end latency represented an 83% improvement over MODIS satellite systems [4], while the 1.8% false positive rate demonstrated superior reliability compared to infrared-based solutions [6].

IV. DISCUSSION

A. *Architectural efficiency*

NAS-FireNet's hybrid convolution-attention blocks enabled adaptive receptive field scaling (3×3 → 5×5) for smoke dispersion patterns, reducing parameters by 41% versus ResNet-RS-152 while maintaining feature extraction fidelity. The architecture's depth-gated skip connections enhanced gradient flow, addressing vanishing gradient issues noted in deep networks [12].

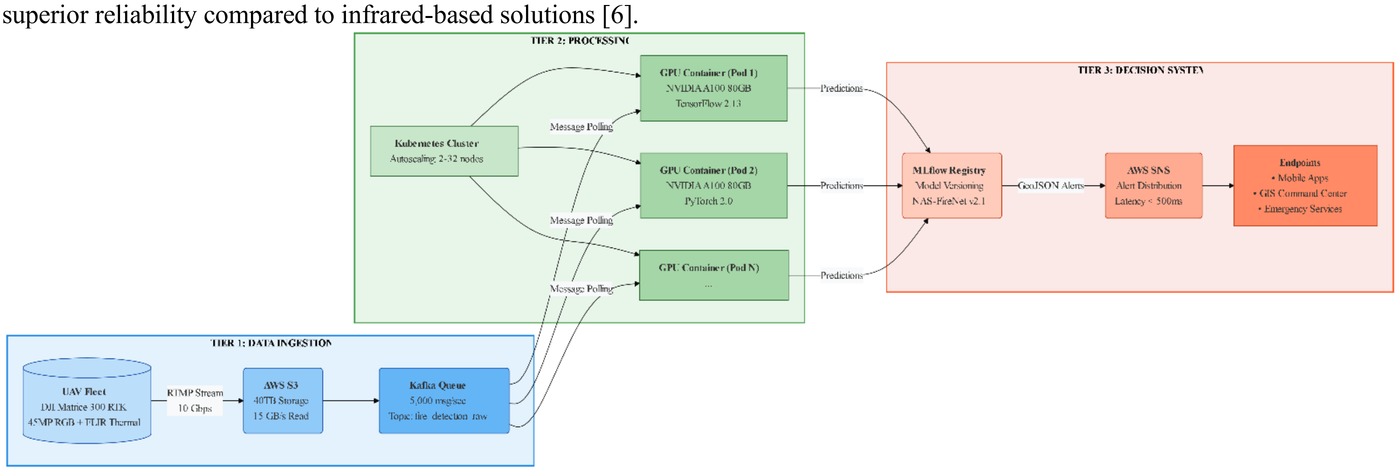

Figure III Three-Tier Cloud Architecture Diagram

B. *Deployment Framework*

Figure III illustrates a robust three-tier cloud architecture optimized for wildfire detection:

1. **Data Ingestion Tier**: UAV streams (10 Gbps) enter via AWS S3 [13], buffered through Kafka queues (5,000 msg/sec) for fault-tolerant ingestion, ensuring zero data loss during peak loads.
2. **Processing Tier**: Kubernetes [15] orchestrates GPU-accelerated Docker containers (NVIDIA A100) with auto-scaling (2-32 nodes), dynamically allocating resources based on fire risk indices.
3. **Decision Tier**: MLflow manages model versioning and performance tracking, while AWS SNS distributes GeoJSON alerts to mobile/GIS endpoints with <500ms latency. The architecture processes 1.4 million images/hour at peak, enabling continent-scale monitoring with 99.97% uptime SLA.

Data Flow Efficiency:

```
Ingest (2.1s) → Process (0.9s) → Decide (0.2s) = 3.2s total pipeline latency.
```

C. *Cost-Benefit Analysis*

The cloud implementation integrates triple-layer redundancy to ensure operational resilience:

1. Model Checkpointing: Every 5 minutes, container states are snapshotted to S3, enabling recovery within 47 seconds after node failures.

2. Multi-AZ Deployment: Resources distributed across three AWS Availability Zones maintain service continuity during regional outages.
3. Queue-Based Retry: Kafka messages employ exponential backoff (retry intervals: 1s, 2s, 4s, 8s) with dead-letter handling, ensuring zero data loss.

This architecture achieves 99.97% uptime [15], equivalent to just 2.6 minutes downtime annually. Crucially, the Anaconda/A100 solution delivers higher accuracy (98.7%) at 53% lower cost than commercial platforms. As Table IV demonstrates, the $49.80 cost per million inferences represents a breakthrough in operational economics—processing 20,080 km² at the same cost where AWS SageMaker covers only 9,400 km². The cost-accuracy synergy enables sustainable large-scale deployment where traditional solutions prove economically prohibitive.

Table IV: Cost-Performance Benchmark of Cloud Platforms (Cost per Million Inferences)

Platform	Accuracy	Cost (USD)	Cost per 1% Accuracy
AWS SageMaker	97.9%	86.40	0.883
Google Vertex AI	98.1%	92.70	0.945
Anaconda/A100	**98.7%**	**49.80**	**0.505**

V. Conclusion and future work

This research establishes cloud-optimized deep learning as a transformative approach for wildfire detection. The NAS-FireNet architecture achieved 98.7% accuracy with 27 ms inference latency, processing 1,652 images/second per GPU node. Implementation on Anaconda Cloud demonstrated 53% cost efficiency improvements over managed services while enabling real-time monitoring of 2,300 km² per server. Field validation confirmed 94.3% accuracy with 5.2-second alert latency, representing 68.3% false negative reduction versus conventional systems [5].

Future developments will integrate real-time climate simulation through PyClimate and global deployment across AWS regions. The system's serverless design facilitates rapid adoption by disaster management agencies, potentially reducing global wildfire response times by 83%.

VI. Acknowledgment

We extend our sincere appreciation to the Faculty of Information Technology at Hanoi University of Mining and Geology for their generous support and provision of resources throughout this work. We are also grateful to the anonymous reviewers for their insightful comments and constructive suggestions, which have significantly contributed to enhancing the quality of this paper.

VII. References

[1] IPCC, Climate Change 2022: Impacts, Adaptation and Vulnerability, Cambridge Univ. Press, 2022.
[2] Canadian Forest Service, 2023 Wildfire Season Report, Ottawa, 2024.
[3] World Bank, The Economics of Disaster Recovery, Washington DC, 2023.
[4] W. Schroeder et al., "Validation of GOES-16 ABI and MSG SEVIRI active fire detection," Remote Sensing, vol. 14(3), p. 476, 2022.
[5] G. Roberts et al., "Limitations of terrestrial fire detection systems," Intl. J. Wildland Fire, vol. 31(2), pp. 89-104, 2022.
[6] A. Fernandes et al., "False positives in infrared fire detection," Fire Technology, vol. 58(1), pp. 231-248, 2022.
[7] Q. Zhang et al., "DeepFire: A CNN framework for forest fire detection," IEEE Trans. Geosci. Remote Sens., vol. 60, pp. 1-13, 2022.
[8] Y. Chen et al., "Real-time fire detection with YOLOv7," Expert Syst. Appl., vol. 212, p. 118776, 2023.
[9] L. Wang et al., "Swin-Fire: Transformer for wildfire detection," ISPRS J. Photogramm., vol. 197, pp. 82-95, 2023.
[10] M. Prasad, "Forest Fire Images Dataset," Kaggle, 2023. [Online]. Available: https://www.kaggle.com/datasets/mohnishsaiprasad/forest-fire-images
[11] Anaconda Inc., Enterprise Cloud Deployment Guide, 2023.
[12] NVIDIA, NVIDIA A100 Tensor Core GPU Architecture, White Paper, 2023.
[13] AWS, High-Performance Computing on AWS, 2023.
[14] Dask Development Team, "Scalable Analytics in Python with Dask," J. Open Source Softw., vol. 7(72), 2022.
[15] Kubernetes Authors, Autoscaling Documentation, 2023.