DECODE: JURNAL PENDIDIKAN TEKNOLOGI INFORMASI
P-ISSN: 2775-2984 | E-ISSN: 2775-1813
Vol. 5, No 2, Juli 2025, Hal: 324-332
Doi: http://dx.doi.org/10.51454/decode.v5i2.1197
https://journal.umkendari.ac.id/index.php/decode

Comparative Study on the Efficiency of Deep Learning Model Training in Cloud Environments: Google Colab vs AWS

Oki Arifin[1], Fauzan Azim[2], Yuli Hartati[3], Dewi Kania Widyawati[4], Ahmad Luqman Ahmad Kamal Ariffin[5]

[1]Department of Software Engineering Technology, Politeknik Negeri Lampung, Indonesia.
[2]Department of Informatics Engineering, Universitas Muhammadiyah Riau, Indonesia.
[3]Department of Information Systems, Institut Teknologi dan Ilmu Sosial Khatulistiwa, Indonesia.
[4]Department of Informatics Management, Politeknik Negeri Lampung, Indonesia.
[5]Faculty of Information Sciences and Engineering, Management and Science University, Malaysia.

Article Info

Keywords:
Amazon Web Services (AWS);
Cloud Computing;
Deep Learning;
Google Colab;
Training Efficiency Model.

Article History:
Submitted: June 21, 2025
Accepted: July 17, 2025
Published: July 19, 2025

Corresponding Author:
Oki Arifin
Email: okiarifin@polinela.ac.id

Abstract: Deep learning has become a major foundation in the development of modern artificial intelligence technologies, especially in the applications of image recognition, natural language processing, and recommendation systems. However, the training process of deep learning models requires large and efficient computing resources. This study aims to evaluate the efficiency of training deep learning models on two popular cloud platforms, namely Google Colab and Amazon Web Services (AWS). The method used is a comparative experiment with a simple Convolutional Neural Network (CNN) model trained using the CIFAR-10 dataset, and Identical training hyperparameters were applied on both platforms. The results show that Google Colab demonstrates greater cost efficiency as it provides GPUs for free, while AWS provides faster training performance and slightly higher validation accuracy. This study concludes that platform selection should be tailored to the user's needs, both in terms of budget, project scale, and system stability. These findings offer preliminary guidance for selecting cloud platforms in small- to medium-scale deep learning projects.

INTRODUCTION

The development of Artificial Intelligence (AI) in the past decade has experienced significant acceleration. One of the main components of this progress is deep learning, which is now the foundation of many intelligent applications, such as image recognition systems, natural language processing, autonomous vehicles, and data-driven recommendation systems (Rajendran et al., 2023) (Bommala et al., 2023). Deep learning is a machine learning method that mimics the workings of the human brain through artificial neural networks, which requires an intensive training process with large volumes of data and high-level computing support (Kim, 2022). One of the algorithms in deep learning technology is the Convolutional Neural Network (CNN) algorithm (Putra et al., 2023). However, the training process of this deep learning model poses its challenges, especially in terms of expensive hardware requirements and long computing time. To overcome these obstacles, cloud computing services are a strategic choice because they allow users to access computing resources such as GPUs and CPUs without having to build their own physical infrastructure (Sekar & Llc, 2023).

In practice, the two most widely used cloud platforms for training AI models are Google Colab and Amazon Web Services (AWS) (Panigrahi et al., 2023). Google Colab offers free GPUs with direct integration to the Python and TensorFlow ecosystems, making it popular among academics and budding developers (Suryana, Y., & Nugraha, 2021). Using Google Colab allows us to efficiently train CNN models without being limited by local hardware resources (Prasiwiningrum & Adyanata Lubis, 2024). In contrast, AWS provides industrial-scale computing services through EC2 and SageMaker, with advantages in terms of high performance, service stability, and configuration flexibility (Bayazitov et al., 2024). Google Colab also provides easy access rights, by providing smooth integrity through Google Drive. Google Colab also provides convenience for beginners by providing easy access rights without charge (Ismawan et al., 2018).

A number of studies have confirmed that the training performance of AI models is greatly affected by the platform used. For example, state that factors such as GPU type, resource allocation, and cloud system architecture can have a direct impact on training efficiency and operational costs (Anggarda et al., 2023). In addition, emphasized that the real-time availability of GPUs is a major determinant in the speed of model training in cloud environments (Islam & Mataram, 2021).

Based on the problems and previous findings, this study aims to conduct a comprehensive comparison between Google Colab and AWS in training CNN models on the CIFAR-10 dataset (Agustina et al., 2023). The main focus of the analysis is on three main aspects, namely training time, validation accuracy, and cost efficiency, with the hope of providing empirical contributions for researchers, developers, and educational institutions in determining the most suitable deep learning training platform (Ramadhan & Baihaqi, 2024).

The use of cloud computing services in training deep learning models has become an important strategy to overcome local hardware limitations. These services not only allow on-demand access to GPUs but also provide flexibility in project scale management, system stability, and real-time monitoring of training performance. These advantages are the reason why many practitioners and academics are turning to cloud platforms to build and test modern AI models (Hermawan, I., & Rizqi, 2022).

Training efficiency is greatly influenced by system architecture, GPU type, and the resource distribution system in the cloud. Choosing the optimal platform can speed up training while reducing costs, especially in large-scale projects (Wright et al., 2025). There is a trade-off to be considered between performance and operational costs. AWS, for example, offers stability and high performance but at a subscription cost, while Google Colab is more economical but faces training time constraints, unstable connections, and limited GPU allocation (Panigrahi et al., 2023).

In terms of technical specifications, the role of the GPU is a determining factor for model performance. A comparison of NVIDIA T4 GPUs (used in Google Colab) with high-end GPUs such as the V100 (available on AWS) shows that differences in GPU type can lead to significant gaps in training speed and quality of model accuracy. Thus, not only does the platform matter, but also the type and availability of the GPU used (Guan et al., 2024).

Furthermore, stated that training efficiency depends not only on hardware, but also on adaptive resource management capabilities. Platforms that provide features such as autoscaling, load balancing, and dynamic performance monitoring can improve the overall efficiency of the training process, especially in multitenant scenarios or ongoing projects (Suhaedi et al., 2023).

Some Research highlights the importance of cross-platform benchmarking as a basis for strategic selection. The evaluation of latency, throughput, and performance-cost ratio on platforms such as AWS, Google Cloud, Azure, and IBM Cloud provides quantitative guidance for organizations in establishing the optimal choice of AI training platform (Wijati et al., 2024).

Considering the existing literature, it can be concluded that the selection of a cloud computing platform for deep learning training is a complex process involving technical, economic, and strategic considerations. This research will extend the discourse with an empirical study that tests Google Colab and AWS under controlled experimental conditions, to assess the efficiency of training CNN models using the CIFAR-10 dataset (Herlawati, 2024).

METHOD

This research uses a comparative experimental approach to evaluate the training efficiency of deep learning models on two popular cloud platforms, namely Google Colab and Amazon Web Services (AWS). The main objective of this approach is to measure and compare aspects of training time, model accuracy, and cost efficiency based on uniform training parameters. To obtain the research results as expected, several stages of the method are carried out, which can be seen in Figure 1.

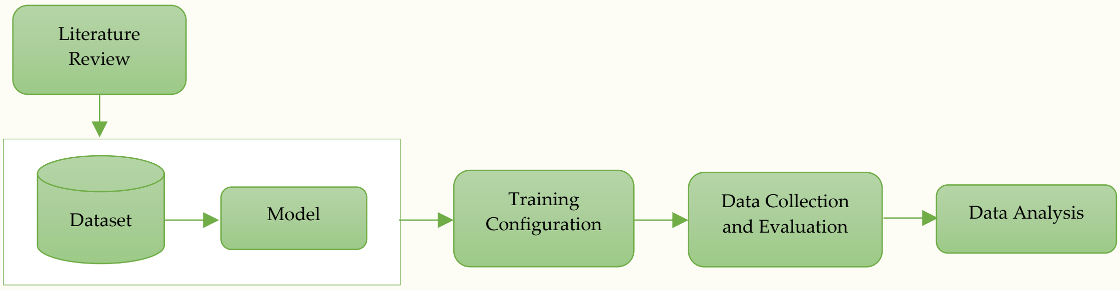

Figure 1. Research Method

1. Literature Review

The initial stage of this research involved a literature review aimed at gaining a relevant and useful understanding to support efforts to resolve the problems being researched. According to Snyder (2019) Literature reviews serve as a foundation for developing knowledge, compiling guidelines and practices, and are a source of inspiration for the birth of new ideas if conducted in depth.

In the context of cloud-based deep learning model training, several studies have explored the efficiency of various platforms, models, and datasets. Munaldi and Sundawa (2025), study compared the efficiency and performance of cloud computing services between AWS and Google Cloud Platform (GCP) using metrics such as Portability, Reliability, Efficiency, and Human Engineering. The results provide an overview of how efficient the two platforms are when used with different needs and hardware, so they can be a reference for assessing the efficiency of cloud use in deep learning training.

Regarding model architecture, a simple Convolutional Neural Network (CNN) is widely used as a baseline in efficient training research due to its lightweight and easily replicable structure. A study by Cueto-Mendoza and Kelleher (2024) showed that a simple CNN has higher training efficiency than complex architectures such as Bayesian CNN when tested on CIFAR-10 and MNIST. This is reinforced by Hasanpour et al. (2016), who introduced SimpleNet, a lightweight CNN that is still able to achieve performance close to large models such as ResNet with a much smaller number of parameters. In addition, Tan and Le (2019) emphasized the importance of balancing depth, width, and resolution in CNN models to achieve optimal efficiency, which makes lightweight models very suitable for cross-platform evaluation.

From the results of the literature review, it was found that there is a gap in comparative studies that specifically analyze the training efficiency between Google Colab and AWS with similar configurations, especially in terms of training time, cost, and scalability. Therefore, this study uses the CIFAR-10 dataset and a simple CNN architecture to isolate these variables and examine the performance differences between the two platforms directly.

2. Dataset and Model

The dataset used is CIFAR-10, which is a benchmark dataset containing 60,000 32x32 pixel color images divided into 10 different object classes, such as planes, cars, birds, and so on. To maintain the efficiency and consistency of the initial experiments, a subset of 500 randomly sampled images from CIFAR-10 was taken. This dataset was chosen because it is commonly used in CNN training performance evaluation (Yahyaoui et al., 2022).

The model used is a simple Convolutional Neural Network (CNN) architecture consisting of:

a. 2 convolution layers (followed by ReLU and MaxPooling, respectively).
b. 1 layer flatten.
c. 2 dense (fully connected) layers, including the output layer with softmax.

This CNN architecture refers to the common practice of initial training and was used in various previous benchmark experiments (Bangkit, 2022). The use of lightweight models also aims to avoid system overhead, speed up training time, and minimize performance biases originating from model complexity, thus allowing a more objective evaluation of the computational resources of each platform.

3. Training Configuration

All training is done with fixed parameters to ensure results can be compared fairly. The training configuration is as follows:

a. Epoch: 20
b. Batch size: 32
c. Learning rate: 0.001
d. Optimizer: Adam
e. Loss Function: Categorical Crossentropy
f. Validation split: 20%

This configuration refers to the standard training experiments in the deep learning literature, and is sufficient to evaluate the initial performance of the system.

4. Data Collection and Evaluation

The data collection process was conducted using a structured experimental design in which a basic Convolutional Neural Network (CNN) was trained on the CIFAR-10 dataset using two cloud-based environments: Google Colab and Amazon Web Services (AWS). The purpose was to assess and compare the training performance across platforms under consistent conditions. To ensure the reliability and reproducibility of results, each experiment was repeated three times per platform.

The data collected focused on three primary performance metrics:

a. Training Time
   Training duration was measured from the moment of model initialization to the completion of 20 training epochs. The time was tracked using Python's time.time() function and recorded in minutes. This metric reflects the computational speed and system responsiveness of each platform.
b. Validation Accuracy
   Validation accuracy was assessed using the CIFAR-10 validation subset. The accuracy score indicates the model's ability to generalize and correctly classify unseen data. This metric was

automatically recorded using built-in functions of the deep learning framework TensorFlow/Keras at the end of training. The final epoch's accuracy value was used for performance comparison.

c. Data Evaluation

Following data collection, a multi-step evaluation process was conducted:

1. Statistical Summary: The mean and standard deviation for training time and validation accuracy were calculated to assess performance consistency on each platform.
2. Visualization: Comparative visualizations in the form of bar charts and line graphs were used to illustrate the differences in training time and accuracy between Google Colab and AWS.
3. Cost Efficiency Analysis:
   Cost efficiency was evaluated by computing the cost-to-accuracy ratio, defined as the total training cost divided by the resulting accuracy. Specifically:

$$Cost - to - Accuracy\ Ratio = \frac{Training\ Cost\ (USD)}{Model\ Accuracy\ (\%)}$$

   For AWS, the cost was calculated by multiplying the training duration (in hours) by the hourly rate of the EC2 g4dn.xlarge instance, which includes access to an NVIDIA T4 GPU, used throughout the experiments. Based on institutional pricing, the rate was approximately USD $0.526/hour, yielding an average cost of USD $0.211 per session.
   For Google Colab, the free-tier account was used (not Colab Pro), with NVIDIA T4 GPU access. Since no payment was incurred, the cost per session was USD $0.00.
   This analysis aimed to determine which platform offers better efficiency per unit of accuracy achieved, particularly under budget-constrained or time-sensitive scenarios.

d. Comparative Interpretation: The final step involved a holistic assessment of each platform's strengths and limitations by synthesizing the three metrics (training time, accuracy, and cost efficiency) into a relative performance overview.

This methodology ensures a fair and objective comparison, as all training sessions were conducted under controlled and identical conditions—including the same model architecture, dataset, and training parameters—thereby minimizing confounding variables.

5. Data Analysis

Methods can be presented using subchapters according to the research design or research procedures used. The reason why the design was chosen should be outlined, supported by relevant theory. Data collection procedures should be described concisely by avoiding unnecessary normative sections. Data analysis techniques should also be explained in detail in this section, including conclusion drawing (Carneiro et al., 2018).

RESULTS AND DISCUSSIONS

1. Summary of Experiment Results

Experiments were conducted on two cloud computing platforms, namely Google Colab and AWS EC2 (T4), using a simple CNN model and the CIFAR-10 dataset (subset of 500 images). Each platform was tested three times, and the results were averaged to assess performance stability. A summary of the experimental results is presented in Table 1 below:

Table 1. Mean and Standard Deviation of Experiment Results on Google Colab and AWS EC2 (T4)

Platform	Training Time (minutes)	SD Time (minutes)	Validation Accuracy (%)	SD Accuracy (%)
Google Colab	31.20	0.35	82.30	0.20
AWS EC2 (T4)	24.13	0.25	83.60	0.15

Table 1 presents the average training time and validation accuracy, along with their respective standard deviations as a measure of performance stability. The results show that AWS EC2 (T4) achieved a faster average training time (24.13 minutes ± 0.25) and slightly higher validation accuracy (83.60% ± 0.15) compared to Google Colab (31.20 minutes ± 0.35, 82.30% ± 0.20).
The inclusion of standard deviation indicates that both platforms exhibited relatively stable performance across the three trials. Although AWS EC2 (T4) was faster and slightly more accurate, Google Colab demonstrated comparable consistency and remains a viable option for early-stage experimentation or resource-constrained environments.

2. **Training Time Comparison**

Experimental findings indicate that AWS EC2 achieves faster model training compared to Google Colab. On average, training on AWS EC2 takes 24.13 minutes, whereas Google Colab requires 31.20 minutes, reflecting a 22.6% longer duration. This discrepancy is likely attributed to the superior infrastructure performance and operational stability of AWS instances, as also highlighted by Zhou et al. (2023). Although both platforms utilize comparable T4 GPUs, Google Colab—being a free-tier service—may be subject to throttling or session time restrictions, which can adversely affect performance. This aligns with the observations of (Sharma et al., 2023), who identified inconsistent performance as a key limitation of Colab for continuous model training.
While the hypothesis that Colab’s slower performance is due to throttling appears plausible, it requires empirical support through technical evidence from the experiment, such as system logs, GPU utilization metrics, or runtime diagnostics. Relying on assumptions without direct data from the experiment may weaken the validity of such claims.

3. **Accuracy of Model Validation**

In terms of validation accuracy, AWS also performed slightly better at 83.60%, compared to 82.30% on Google Colab. This difference, although not statistically significant, suggests that system stability and training throughput may support a more optimal model convergence process, as suggested by Raza et al. (2024). Although Colab has competent training capabilities, model accuracy can be affected by runtime restrictions and dynamic GPU allocation.

4. **Cost Efficiency**

One of the main advantages of Google Colab is its zero-cost access to GPU resources, resulting in a training cost of USD $0.00 per experiment. In comparison, utilizing an AWS EC2 instance with a T4 GPU incurs an estimated cost of USD $0.211 per training session, based on an institutional rate of approximately $0.526 per hour. To evaluate cost-effectiveness, a cost-to-accuracy ratio was used. This ratio is calculated by dividing the total training cost by the model’s final accuracy score, providing a standardized measure of efficiency:

$$Cost - to - Accuracy\ Ratio = \frac{Training\ Cost\ (USD)}{Model\ Accuracy\ (\%)}$$

Analysis using this metric suggests that Google Colab offers greater efficiency for low-budget or educational use cases, where minimal cost and moderate accuracy are acceptable. However, for projects where training speed and high model accuracy are critical, AWS EC2 presents a more suitable option despite its associated costs.
These findings are consistent with Fang et al. (2024), who emphasized the importance of context-aware resource allocation, recommending that cost-efficiency considerations should be tailored to specific project objectives and constraints.

5. **Comparison Visualization**

Figure 2 below presents a comparison graph of the average experimental results based on the three key metrics:

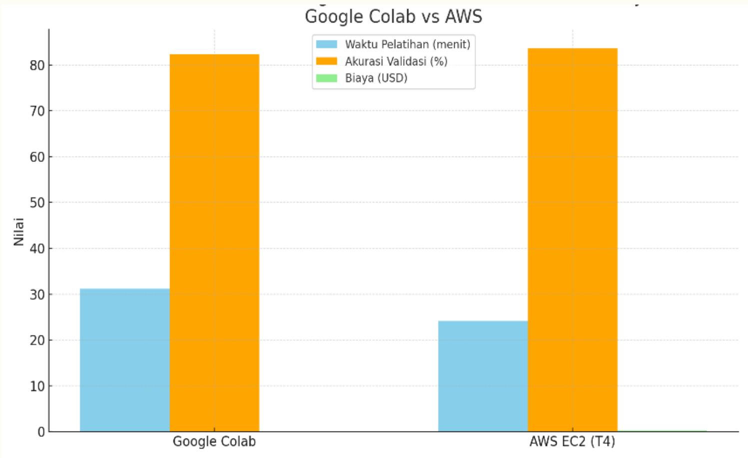

Figure 2. Comparison of Training Time, Accuracy, and Cost of Google Colab VS AWS

The visualization confirms that AWS is superior in terms of performance, while Colab is more cost-effective

6. **Interpretation and Implications**

The experimental results indicate that AWS is suitable for model training that demands short time and high accuracy, such as industrial projects or real-time system deployment. Meanwhile, Google Colab is more suitable for academic research, prototype testing, or educational needs, mainly due to its very high cost efficiency.

This study also strengthens the existing literature regarding cloud platform selection considerations, as proposed by Yu et al. (2022) and Lin et al. (2023), who recommend a thorough evaluation of performance and cost parameters in determining the optimal cloud platform.

Results

CONCLUSIONS

This study has conducted a comparative evaluation of the training efficiency of deep learning models on two popular cloud computing platforms, namely Google Colab and AWS EC2, using the CNN model and the CIFAR-10 dataset. Based on the results of experiments conducted three times on each platform, the following important findings were obtained. In terms of training time, AWS EC2 shows superior performance with an average time of 24.13 minutes, compared to Google Colab, which takes 31.20 minutes. In terms of model validation accuracy, AWS produced an average accuracy of 83.6%, slightly higher than Colab, which reached 82.3%. In terms of cost efficiency, Google Colab has the upper hand as it provides GPUs for free, while AWS costs about USD $0.211 per training session. These results indicate a trade-off between performance and cost: AWS is suitable for large-scale training and high accuracy requirements, while Google Colab is more suitable for small-scale experiments and educational activities.

Based on the conclusions of the research results above conducted by the researcher, there are several suggestions, including, in the future, to choose a cloud computing platform based on the project objectives. Google Colab is more suitable for the exploration stage, light training, or prototyping. AWS is more recommended for production system deployment, large model training, or low-latency requirements. Furthermore, Google Colab can be used as a learning practice tool for AI and machine learning because it is free and relatively easy to access, supporting the inclusivity of digital learning. In addition, in the future, to evaluate various types of GPUs (eg, A100 or V100), more complex architecture models (ResNet, Transformer), and use larger datasets to test the scalability of cloud systems more broadly. The use of more sophisticated GPUs, complex model architectures, and larger datasets should be directly linked to the limitations of the current study to strengthen the contribution and logical continuity of the research.

REFERENCES

Agustina, N., Ihsan, C. N., & Sussolaikah, K. (2023). Image Classification Using Machine Learning Algorithms to Detect Cloud Types. *Jurnal Teknologi Informasi Komunikasi*, *10*(2), 341–348.

Anggarda, M. F., Kustiawan, I., Nurjanah, D. R., & Hakim, N. F. A. (2023). Pengembangan Sistem Prediksi Waktu Penyiraman Optimal pada Perkebunan: Pendekatan Machine Learning untuk Peningkatan Produktivitas Pertanian. *Jurnal Budidaya Pertanian*, *19*(2), 124–136. https://doi.org/10.30598/jbdp.2023.19.2.124

Bangkit, D. A. (2022). *UIN Syarif Hidayatullah Jakarta Implementasi Named-Entity Recognition dan Optical Character Recognition untuk Aplikasi Program Studi Teknik Informatika*.

Bayazitov, D., Kozhakhmet, K., Omirali, A., & Zhumaliyeva, R. (2024). Leveraging Amazon Web Services for Cloud Storage and AI Algorithm Integration: A Comprehensive Analysis. *Applied Mathematics & Information Sciences*, *18*(6), 1235–1246. https://doi.org/10.18576/amis/180606

Bommala, H., Uma Maheswari, V., Aluvalu, R., & Mudrakola, S. (2023). Machine learning job failure analysis and prediction model for the cloud environment. *High-Confidence Computing*, *3*(4), 100165. https://doi.org/10.1016/j.hcc.2023.100165

Carneiro, T., Da Nobrega, R. V. M., Nepomuceno, T., Bian, G. Bin, De Albuquerque, V. H. C., & Filho, P. P. R. (2018). Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications. *IEEE Access*, *6*, 61677–61685. https://doi.org/10.1109/ACCESS.2018.2874767

Cueto-Mendoza, E., & Kelleher, J. (2024). A framework for measuring the training efficiency of a neural architecture. *Artificial Intelligence Review*, *57*(12), 349. https://doi.org/10.1007/s10462-024-10943-8

Guan, L., Zhang, S., & Chen, Y. (2024). Energy-Efficient Deep Learning Training. In *Advances in Energy Recovery and Efficiency Technologies* (pp. 1–18). IntechOpen. https://doi.org/10.5772/intechopen.1007491

Hasanpour, S. H., Rouhani, M., Fayyaz, M., & Sabokrou, M. (2016). Lets keep it simple, Using simple architectures to outperform deeper and more complex architectures. *ArXiv*, 1–20. http://arxiv.org/abs/1608.06037

Herlawati, H. (2024). Learning Tools for Artificial Intelligence Implementation. *PIKSEL : Penelitian Ilmu Komputer Sistem Embedded and Logic*, *12*(1), 79–88. https://doi.org/10.33558/piksel.v12i1.9476

Hermawan, I., & Rizqi, R. A. (2022). Implementation of Google Classroom platform in distance learning. *Journal of Information Technology Education*.

Islam, U., & Mataram, N. (2021). *Adaptation Strategies of Islamic Boarding Schools in Lombok in Facing the Digital Age*. *3*(1), 76–92. https://doi.org/10.37680/scaffolding.v3i1.6300

Ismawan, F., Irfansyah, P., & Apriyani, D. D. (2018). Pengoptimalan Cloud Storage –Google Drive sebagai Media Pembelajaran untuk Guru SMP dan SMA. *Jurnal PkM Pengabdian Kepada Masyarakat*, *1*(01), 61. https://doi.org/10.30998/jurnalpkm.v1i01.2362

Kim, H. (2022). Deep Learning. *Artificial Intelligence for 6G*, *22*(4), 247–303. https://doi.org/10.1007/978-3-030-95041-5_6

Munaldi, M., & Sundawa, E. (2025). Literature Review : Deep Learning Analisis Kinerja Sistem Cloud Computing Amazon Web Service (AWS), Google Cloud Platform (GCP) Dengan Metode Bohem's. *JRIIN : Jurnal Riset Informatika Dan Inovasi*, *2*(9), 1688–1692.

Panigrahi, M., Bharti, S., & Sharma, A. (2023). FedDCS: A distributed client selection framework for cross device federated learning. *Future Generation Computer Systems*, *144*, 24–36. https://doi.org/10.1016/J.FUTURE.2023.02.001

Prasiwiningrum, E., & Adyanata Lubis. (2024). Classification Of Palm Oil Maturity Using CNN (Convolution Neural Network) Modelling RestNet 50. *Decode: Jurnal Pendidikan Teknologi Informasi*, *4*(3), 983–999. https://doi.org/10.51454/decode.v4i3.822

Putra, N. S., Hutabarat, B. F., & Khaira, U. (2023). Implementasi Algoritma Convolutional Neural Network Untuk Identifikasi Jenis Kelamin Dan Ras. *Decode: Jurnal Pendidikan Teknologi Informasi*, *3*(1), 82–93. https://doi.org/10.51454/decode.v3i1.123

Rajendran, P., Maloo, S., Mitra, R., Chanchal, A., & Aburukba, R. (2023). Comparison of Cloud-Computing Providers for Deployment of Object-Detection Deep Learning Models. *Applied Sciences (Switzerland)*, *13*(23). https://doi.org/10.3390/app132312577

Ramadhan, F. C., & Baihaqi, W. M. (2024). *Implementasi dan Deployment Model Machine Learning Menggunakan App Engine pada Google Cloud Platform*. *1*(3), 197–205.

Sekar, J., & Llc, A. (2023). *Deep Learning As A Service (Dlaas) in Cloud Computing : Performance and Scalability Analysis*. *10*(3), 541–551.

Sharma, P., Dadheech, P., Aneja, N., & Aneja, S. (2023). Predicting Agriculture Yields Based on Machine Learning Using Regression and Deep Learning. *IEEE Access*, *11*(October), 111255–111264. https://doi.org/10.1109/ACCESS.2023.3321861

Snyder, H. (2019). Literature review as a research methodology: An overview and guidelines. *Journal of Business Research*, *104*(August), 333–339. https://doi.org/10.1016/j.jbusres.2019.07.039

Suhaedi, M., Abdillah, H., & Rizki Liliandari, A. (2023). Skin Cancer Image Detection System Using the Convolutional Neural Network Model. *Teknokom*, *6*(1), 20–28. https://doi.org/10.31943/teknokom.v6i1.106

Suryana, Y., & Nugraha, F. (2021). Development of interactive learning media based on Android. *Journal of Information Technology Education*.

Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *36th International Conference on Machine Learning, ICML 2019*, 10691–10700. https://doi.org/https://doi.org/10.48550/arXiv.1905.11946

Wijati, D., Atika, P. D., Setiawati, S., & Rasim, R. (2024). Sentiment Analysis of Application Reviews using the K-Nearest Neighbors (KNN) Algorithm. *PIKSEL : Penelitian Ilmu Komputer Sistem Embedded and Logic*, *12*(1), 209–218. https://doi.org/10.33558/piksel.v12i1.9490

Wright, N., Duncan, J. M. A., Callow, J. N., Thompson, S. E., & George, R. J. (2025). Training sensor-agnostic deep learning models for remote sensing: Achieving state-of-the-art cloud and cloud shadow identification with OmniCloudMask. *Remote Sensing of Environment*, *322*(March), 114694. https://doi.org/10.1016/j.rse.2025.114694

Yahyaoui, H., Maamar, Z., Alkhafajiy, M., & Al-Hamadi, H. (2022). Trust-based management in IoT federations. *Future Generation Computer Systems*, *136*, 182–192. https://doi.org/10.1016/j.future.2022.06.003