International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 5, pp 235-240, May-2026

https://doi.org/10.47001/IRJIET/2026.105033

Deep Learning-Based Logo Authentication Using EfficientNet for Fake Logo Detection

[1]Sakshi Shashikant Patil, [2]Janavi Pramod Shinde, [3]Shreya Sampat Patil, [4]Pallavi Rajanikant Patel, [5]Rijavan A. Shaikh

[1,2,3,4]Department of Computer Engineering, D. N. Patel College of Engineering, Maharashtra, India

[5]Assistant Professor, Department of Computer Engineering, D. N. Patel College of Engineering, Maharashtra, India

E-mail: [1]sakshipatil5436@gmail.com, [2]shindejp14@gmail.com, [3]sp7817288@gmail.com, [4]patelpallavi3005@gmail.com, [5]shaikh.rizwanengg@gmail.com

***Abstract* - The Unique Logo Authentication System is a web-based application that detects whether a logo is genuine or fake. It uses image preprocessing techniques and a Convolutional Neural Network (CNN) model to analyze logo features. The system is developed using Python and Django for efficient backend processing and user interaction. It provides quick and reliable results, helping protect brand identity and prevent logo duplication.**

Keywords: Unique Logo, Logo Authentication System, CNN, Logo Verification.

I. INTRODUCTION

The Unique Logo Authentication System is a web-based application developed to detect fake logos of popular brands such as Adidas, Samsung, Puma, and Twitter. It uses modern technologies including Python, Django, SQLite, HTML, CSS, JavaScript, and Bootstrap. The system provides a smooth and user-friendly interface, ensuring ease of use. Its main goal is to ensure logo authenticity and protect brand identity.

The core of the system is based on the EfficientNetB2 architecture, a powerful Convolutional Neural Network (CNN)[1,3] used for image classification. It accurately identifies patterns and features in logos. The model is trained using a dataset of logo images. This improves the system's ability to differentiate between genuine and fake logos.

To enhance model performance, ImageDataGenerator is used for data augmentation. It applies transformations like rescaling, zooming, shearing, and flipping. TensorFlow is used to build and train the model [6]. The system uses EfficientNetB2 with layers like Global Average Pooling and Max Pooling for better feature extraction.

The model is compiled using categorical cross-entropy loss and optimized with the Adam optimizer. Its performance is evaluated using accuracy and confusion matrix. OpenCV is used for image processing tasks such as resizing and prediction display. Pickle helps in saving model data for future use and analysis.

The system consists of two modules: user and admin. The user module allows registration, login, logo prediction, history viewing, and feedback. The admin module manages users, monitors predictions, and handles feedback. Overall, the system provides a reliable and efficient solution for logo authentication.

Unlike traditional logo detection systems, the proposed system focuses on efficient and accurate classification of genuine and fake logos using a lightweight yet powerful EfficientNetB2 model. The integration of deep learning with a user-friendly Django-based web application makes the system practical for real-world use. The approach balances accuracy and computational efficiency, making it suitable for scalable deployment.

II. RELATED WORK

Previous research in logo detection has explored both traditional image processing techniques and deep learning-based approaches. Methods such as RL-LOGO (Fujitake, 2023) [11] focus on accurate logo localization using reinforcement learning, while Panda et al. (2022) [12] proposed techniques for phishing detection using logo identification. Deep learning models like CNNs, ResNet50 [4], and InceptionV3 [2] have shown significant improvements in accuracy. However, these methods often require high computational resources or large datasets. This highlights the need for efficient models like EfficientNetB2 [1] that provide a balance between accuracy and computational cost.

2.1 Traditional Image Processing Method

Earlier logo detection methods used techniques like edge detection, feature extraction, and template matching. These approaches worked well for simple logos but struggled with complex variations and noisy images. As a result, their accuracy was limited in real-world scenarios.

ISSN (online): 2581-3048
Volume 10, Issue 5, pp 235-240, May-2026
https://doi.org/10.47001/IRJIET/2026.105033

2.2 CNN Based Logo Recognition

CNN-based methods use deep learning to automatically extract important features from logo images. They provide higher accuracy compared to traditional techniques and perform well on complex data. However, they require large datasets and high computational resources.CNN is widely used for image classification tasks [3] because it automatically extracts important visual features such as edges, shapes, colors, and textures from images. In logo recognition, CNN helps identify patterns in genuine and fake logos. However, basic CNN models may require more training time and may not always provide the highest accuracy for complex logo datasets.

2.3 Advanced Deep Learning Model

Advanced models like EfficientNetB2 provide high accuracy with fewer parameters. Techniques such as data augmentation and transfer learning improve performance and generalization. However, challenges like similar logos and varying image quality still remain.

- **ResNet50**

ResNet50 is a deep learning model that uses residual connections to solve the vanishing gradient problem in deep networks. It can learn more detailed features from logo images and provides better accuracy than traditional CNN models. However, ResNet50 is computationally heavier and requires more training time.[4]

- **InceptionV3**

InceptionV3 is an advanced deep learning model that uses multiple filter sizes to capture logo features at different scales. It performs well in recognizing logos with varying shapes, textures, and complex backgrounds. It provides high accuracy but needs more computational resources compared to simpler CNN models.[2]

- **EfficientNetB2**

EfficientNetB2 is an optimized deep learning model that balances accuracy and computational efficiency. It uses fewer parameters while still providing excellent performance on large image datasets. Due to its faster training speed, lower memory usage, and high accuracy, EfficientNetB2 is selected as the main model for the proposed system.[1]

III. PROPOSED METHODOLOGY

The proposed methodology for the Unique Logo Authentication System is designed to provide a reliable, automated, and highly accurate solution for distinguishing genuine logos from fake ones. The system integrates deep learning, image preprocessing, and a scalable Django-based web interface to deliver efficient performance and ease of use. The process begins with data collection, where genuine and fake logo images from brands such as Adidas, Samsung, Puma, and Twitter are gathered. These images undergo preprocessing, including resizing to a standard dimension required by EfficientNetB2, normalization, and noise removal using OpenCV. The dataset is then split into training, validation, and testing sets, and further enhanced using data augmentation techniques such as rotation, zooming, flipping, shearing, and rescaling to increase model robustness.

For model development, the EfficientNetB2 architecture is employed due to its high accuracy and computational efficiency. The model is customized by adding layers such as Global Average Pooling, Global Max Pooling, a concatenation layer, dense layers, and a softmax classifier. Training is performed using the Adam optimizer and categorical cross-entropy loss, with accuracy and loss values recorded for evaluation. The model's performance is validated using accuracies, prediction probabilities, and confusion matrix results, and only the best-performing weights are saved for deployment.

The system processes uploaded logo images using OpenCV for preprocessing and an EfficientNetB2 model for prediction, generating class labels and probability scores. Results are stored in a SQLite database, while the Django web app [7] provides user features like upload, history, and feedback, and admin features for monitoring and management. The model is integrated into Django, deployed on a server, and continuously improved through user feedback and periodic retraining.

IV. EXPERIMENTAL SETUP

The original dataset consists of 167,140 logo images across 2341 classes. However, for the proposed system, a subset of 2328 images was selected, focusing on 4 major brands (Adidas, Puma, Samsung, and Twitter) along with their counterfeit versions. This selection was made to reduce computational complexity and enable more focused classification. The dataset was divided into training, validation, and testing sets to ensure proper evaluation of the model.

The system is implemented using Python libraries such as TensorFlow, Keras, OpenCV, NumPy, Pandas, and Django. The model is trained using the Adam optimizer and categorical cross-entropy loss function. The performance of the system is evaluated using metrics such as accuracy, precision, recall, F1-score, and confusion matrix. Class-wise performance is analyzed using the confusion matrix.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 5, pp 235-240, May-2026

https://doi.org/10.47001/IRJIET/2026.105033

Additionally, model performance can be further assessed using ROC curves and Precision–Recall curves. Cross-validation techniques may also be applied to improve model robustness.

Table 1: Hardware/Software Details

Software/Hardware	Details
Operating System	Windows
Programming Language	Python
Framework	Django
Database	SQLite
Deep Learning Libraries	TensorFlow, Keras
Image Processing Library	OpenCV
Supporting Libraries	NumPy, Pandas
Processor	Intel i5 or above
Ram	8 GB

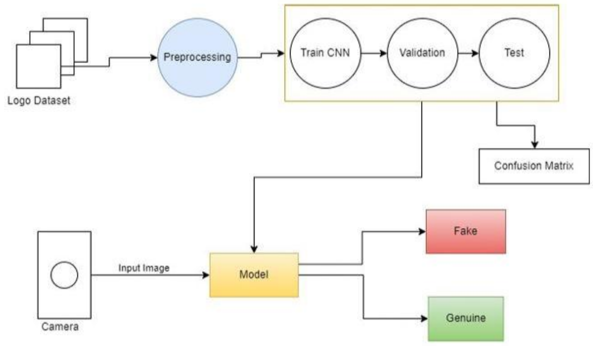

Figure 1: System Architecture Diagram

The system architecture includes user image upload, image preprocessing, EfficientNetB2-based classification, prediction result generation, and database storage.

Figure 2: Home Page

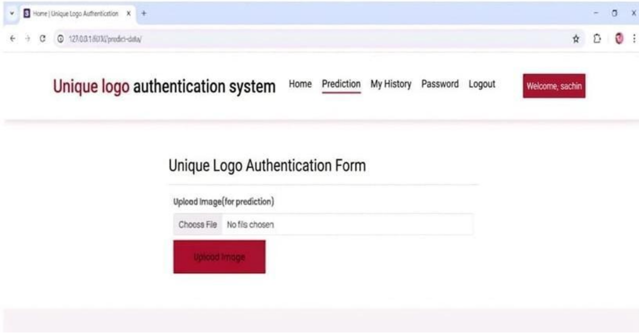

Figure 3: Perform Prediction Page

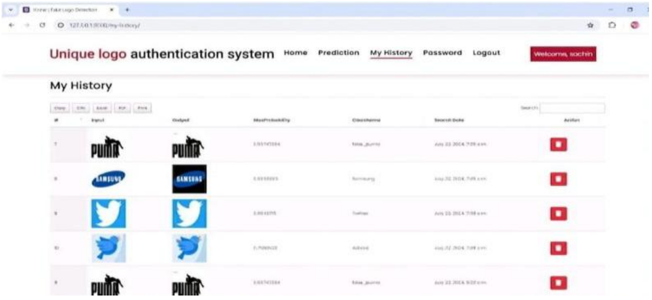

Figure 4: View Prediction Result History Page

V. RESULTS AND DISCUSSIONS

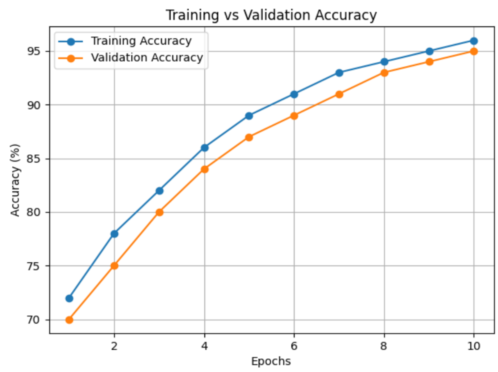

Figure 5: Training vs Validation Accuracy Graph

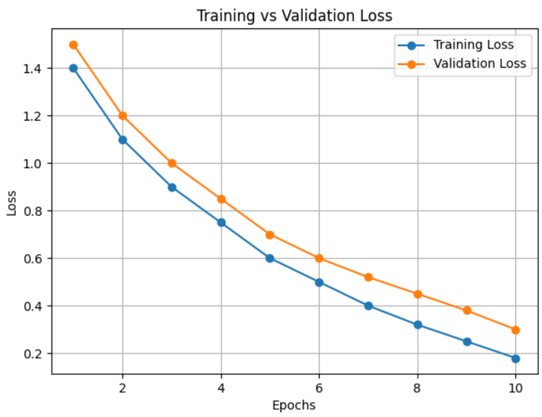

Figure 6: Training vs Validation Loss Graph

ISSN (online): 2581-3048
Volume 10, Issue 5, pp 235-240, May-2026
https://doi.org/10.47001/IRJIET/2026.105033

The proposed model achieved 95% validation accuracy and around 96% training accuracy. The accuracy graph shows that both training and validation accuracy increased steadily with epochs. The validation curve remained close to the training curve, indicating that the model did not suffer from major overfitting.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F1 Score} = 2 \times \frac{(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})}$$

The loss graph shows a continuous decrease in both training loss and validation loss. The final validation loss remained low, which indicates that the model generalized well on unseen logo images.

Among different brands, some logos achieved slightly higher accuracy because of their unique design patterns and shapes. Brands with more similar logo structures showed slightly lower accuracy, but the overall system performance remained strong.

Future improvements may include real-time camera-based logo detection, mobile application support, and comparison with models such as ResNet50 and InceptionV3.

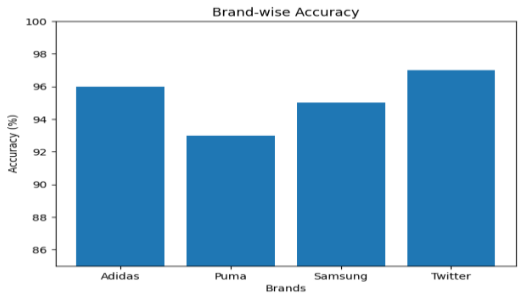

Figure 7: Brand-wise Accuracy Graph

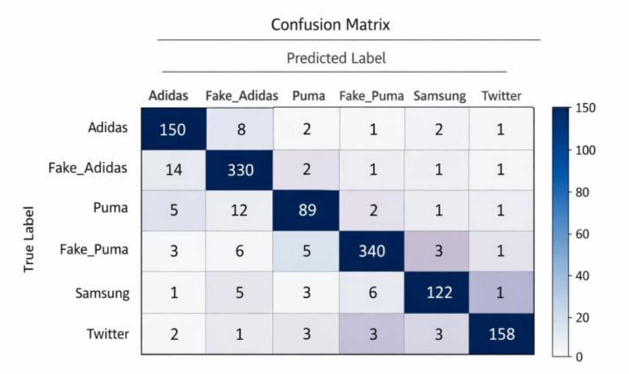

Figure 8: Confusion Matrix of the Proposed Model

The confusion matrix shows the classification performance of the model across different logo classes and helps evaluate prediction accuracy.

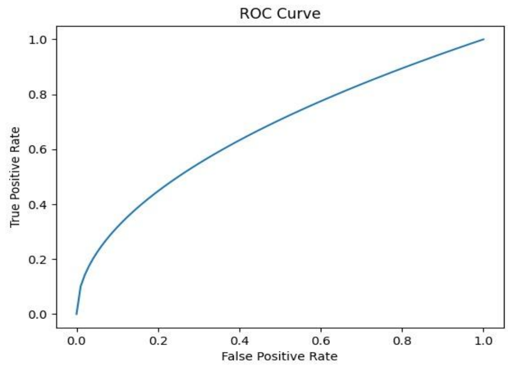

Figure 9: ROC curve graph

The ROC curve represents the trade-off between True Positive Rate and False Positive Rate. The curve indicates that the model achieves a high true positive rate with a low false positive rate, demonstrating strong classification performance.

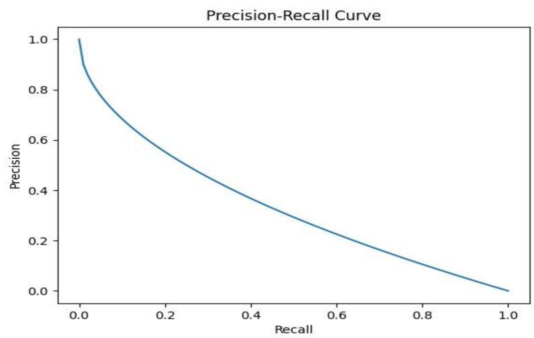

Figure 10: Precision–Recall curve graph

The Precision–Recall curve shows the relationship between precision and recall for the model. The curve indicates that the model maintains high precision across different recall values, confirming its effectiveness in distinguishing genuine and fake logos.

Table 2: Result Table

Metric	Value
Training Accuracy	96%
Validation Accuracy	95%
Training Loss	0.18
Validation Loss	0.30
Dataset Size	2328 Images
Number of Classes	4

ISSN (online): 2581-3048
Volume 10, Issue 5, pp 235-240, May-2026
https://doi.org/10.47001/IRJIET/2026.105033

Table 3: Model Comparison Table

Model	Accuracy	Training Time	Performance
CNN	91%	High	Good
ResNet50	93%	Moderate	Better
InceptionV3	94%	Moderate	Very Good
EfficientNetB2	95%	Low	Excellent

From the comparison, EfficientNetB2 achieved the highest accuracy with lower training time, making it the most suitable model for the proposed system.

VI. CONCLUSION

The Unique Logo Authentication System provides an efficient and automated solution for identifying genuine or fake logos using deep learning and image processing techniques. By integrating CNN-based prediction with a user-friendly web interface, the system eliminates the need for manual verification and reduces human error. It accurately analyzes logo features, classifies them, and delivers instant results with high reliability and speed. The system also includes essential features such as user authentication, preprocessing, dataset management, result history, and admin monitoring, making it a complete end-to-end application. Overall, the project demonstrates how AI can help prevent brand misuse, protect originality, and support industries affected by counterfeiting. The proposed system demonstrates the practical applicability of deep learning in real-world counterfeit detection and brand protection systems.

VII. FUTURE DIRECTION

In future, the Unique Logo Authentication System can be enhanced by integrating real-time camera-based logo detection and mobile application support. The system can also be expanded to include more international brands and larger datasets for better accuracy. Advanced deep learning models such as ResNet50 and InceptionV3 [4,2] can be compared with EfficientNetB2 to improve speed and prediction performance. Future versions may also support e-commerce product verification and counterfeit detection in online marketplaces.

REFERENCES

[1] Tan, M., Le, Q. V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *Proceedings of the 36th International Conference on Machine Learning (ICML).*

[2] Chollet, F. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

[3] Goodfellow, I., Bengio, Y., Courville, A. (2016). Deep Learning. *MIT Press*.

[4] He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep Residual Learning for Image Recognition. *IEEE CVPR*.

[5] Pedregosa, F. et al. (2011). Scikit-Learn: Machine Learning in Python. *Journal of Machine Learning Research*.

[6] Abadi, M. et al. (2015). TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. *Google Research*.

[7] Django Software Foundation. (2023). Django Documentation. Retrieved from https://www.djangoproject.com/

[8] Gonzalez, R. C., Woods, R. E. (2018). Digital Image Processing. *Pearson.*

[9] Zeiler, M. D., Fergus, R. (2014). Visualizing and Understanding Convolutional Networks. *ECCV.*

[10] Kingma, D. P., Ba, J. (2015). Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations (ICLR)*

[11] Liang, Z., et al. (2025). Logo-VGR: Visual Grounded Reasoning for Open-world Logo Recognition. *arXiv*.

[12] Fujitake, M. (2023). RL-LOGO: Deep Reinforcement Learning Localization for Logo Recognition. *arXiv*. 35

[13] Panda, P., et al. (2022). A Novel Logo Identification Technique for Logo-Based Phishing Detection. *MDPI.*

[14] Unknown Authors. (2023). fake logo authentication system using CNN. *AIP Conference Proceedings.*

[15] Rani, V. Y., Balkrishna, R. S. (2025). fake Logo Detection Using Deep Learning. *IJETRM.*

[16] Sadineni, V. V., et al. (2024). Online fake logo authentication system. *EasyChair.*

[17] Jenila, J., et al. (2025). Detect Genuine and fake Logos using Inception-V3. *IJFANS.*

[18] Thilagaraj, T., et al. (2024). A Review of fake logo authentication system using Machine Learning and Deep Learning. *ResearchGate*.

[19] Various Authors. (2025). fake logo authentication system using Generative Deep Learning. *IRJMETS*.

[20] Devane, S., Dhavale, S., Katkar, A. (2025). fake logo authentication system using ORB + FLANN. *MRI India*.

[21] Szegedy, C., et al. (2015). Going Deeper with Convolutions. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* (Inception model–base of Inception-V3).

[22] Redmon, J., et al. (2016). You Only Look Once: Unified, Real Time Object Detection. *IEEE CVPR.* (YOLO– logo detection Localization).

[23] Redmon, J., Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *arXiv*. (Fast logo detection in real-world images)

IRJIET

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 10, Issue 5, pp 235-240, May-2026

https://doi.org/10.47001/IRJIET/2026.105033

[24] Liu, W., et al. (2016). SSD: Single Shot MultiBox Detector. *European Conference on Computer Vision (ECCV).* (Used for logo object detection).

[25] Simonyan, K., Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *ICLR.* (VGG– feature extraction for logos) 36.

[26] Ojala, T., Pietik¨ainen, M., M¨aenp¨a¨a, T. (2002). Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE TPAMI*. (Traditional feature-based logo analysis).

[27] Lowe, D. G. (2004). Distinctive Image Features from ScaleInvariant Keypoints. *International Journal of Computer Vision*. (SIFT– classic logo matching method).

[28] Zoph, B., Le, Q. V. (2017). Neural Architecture Search with Reinforcement Learning. *ICLR*. (RL concept relevant to RL-LOGO).

[29] Krizhevsky, A., Sutskever, I., Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *NIPS.* (AlexNet– CNN foundation).

[30] Zhu, X., et al. (2017). Deep Learning for Logo Detection: A Survey. *IEEE Transactions on Neural Networks and Learning Systems.* (Survey specific to logo detection).

Citation of this Article:

Sakshi Shashikant Patil, Janavi Pramod Shinde, Shreya Sampat Patil, Pallavi Rajanikant Patel, & Rijavan A. Shaikh. (2026). Deep Learning-Based Logo Authentication Using EfficientNet for Fake Logo Detection. *International Research Journal of Innovations in Engineering and Technology - IRJIET*, 10(5), 235-240. Article DOI https://doi.org/10.47001/IRJIET/2026.105033

****