iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

The Visual Similarity Search and Retrieval System Using Mobilenetv2

P. Devi Sravanthi[1], Patta Aswinitha[2], Pathan Adilsha Khan[3], Thadi Vinod[4], Jada Mohan Chandra Ajay[5], Sunkara Venkatesh[6]

[1]*Assistant Professor, Department of Computer Science and Engineering (AI), Pragati Engineering College, ADB Road, Surampalem, Near Kakinada, East Godavari District, Andhra Pradesh, India-533437*

[2,3,4,5,6] *B.Tech Students, Department of Computer Science and Engineering(AI), Pragati Engineering College, ADB Road, Surampalem, Near Kakinada, East Godavari District, Andhra Pradesh, India-533437*

Abstract: ***The exponential growth of digital images across various domains has significantly increased the demand for efficient and intelligent image retrieval systems. Traditional keyword-based retrieval approaches are limited due to dependency on manual annotations and lack of semantic understanding. To address these challenges, this research proposes a deep learning-based visual similarity search system that enables efficient content-based image retrieval using MobileNetV2 and FAISS.The proposed system extracts high-level visual features from images using MobileNetV2, a lightweight convolutional neural network optimized for performance and efficiency. These features are represented as numerical vectors that capture essential characteristics such as color distribution, texture patterns, and object structure. A similarity search mechanism based on cosine similarity is employed to compare feature vectors and identify visually similar images. To enhance scalability and search speed, FAISS (Facebook AI Similarity Search) is integrated, enabling fast nearest-neighbor retrieval even in large datasets. The system is implemented with a web-based interface that allows users to upload query images and retrieve ranked results based on similarity scores. Experimental analysis demonstrates that the proposed system achieves high retrieval accuracy with reduced computational overhead. The framework is scalable, efficient, and suitable for real-world applications such as e-commerce, digital libraries, medical imaging, and multimedia content management systems.***

Keywords: Content-Based Image Retrieval, MobileNetV2, FAISS, Deep Learning, Visual Similarity Search, Feature Extraction, Image Retrieval, Computer Vision.

I. INTRODUCTION

With the rapid expansion of digital media, managing and retrieving relevant images from large datasets has become a significant challenge. Traditional image retrieval methods rely on metadata or textual descriptions, which are often incomplete or subjective. This limitation has led to the development of Content-Based Image Retrieval (CBIR) systems that analyze the visual content of images.The proposed system introduces an advanced visual similarity search framework that utilizes deep learning to extract meaningful image features. MobileNetV2 is employed for its efficiency and ability to generate high-quality feature representations, making it suitable for real-time applications. The system focuses on identifying visually similar images by comparing feature vectors rather than relying on textual annotations. By integrating FAISS, the system ensures fast and scalable similarity search, even with large datasets. Human emotions are a fundamental component of interpersonal communication and behavioral understanding. Facial expressions serve as one of the most expressive channels through which emotional states are conveyed. With the evolution of artificial intelligence and deep learning technologies, there has been significant progress in enabling machines to interpret such expressions automatically.

A. Problem Statement

Existing image retrieval systems depend heavily on textual metadata, which may not accurately represent image content. Additionally, searching large image datasets is computationally expensive and time-consuming. There is a need for an automated, efficient, and scalable system that can retrieve images based on visual similarity.

B. Motivation

The motivation behind this research is to develop a system that mimics human visual perception by identifying similarities between images based on their visual features. This approach enhances retrieval accuracy and reduces dependency on manual annotations. The integration of lightweight deep learning models and efficient search algorithms further improves system performance.

©IJRASET: All Rights are Reserved

C. Key objectives of this research include

The primary objective of this research is to design and develop an efficient visual similarity search system capable of retrieving images based on their visual content using deep learning techniques; to utilize MobileNetV2 for extracting meaningful feature representations from images; to implement an effective similarity measurement technique such as cosine similarity for accurate comparison of image features; to integrate FAISS for fast and scalable nearest-neighbor search in large datasets; to develop a user-friendly web interface for seamless image upload and result visualization; and to ensure the system is scalable, efficient, and suitable for real-world applications such as e-commerce, healthcare, and multimedia search systems.

II. LITERATURE SURVEY

Recent progress in content-based image retrieval (CBIR) has shifted the field from handcrafted descriptors toward deep feature embeddings, metric learning, efficient nearest-neighbor indexing, and explainable visual search. For your project, the most relevant prior work falls into five technical areas: semantic image retrieval, CNN-based feature representation, deep similarity learning, real-time object detection for ROI selection, and visual explanation / scalable indexing. The following table is written in a paper-ready format and can be pasted directly into your manuscript. The findings are paraphrased to keep the section original while remaining faithful to the source papers.

S.No	Citation	Research Focus	Methodology	Key Findings
1	Liu et al., 2007	High-level semantic CBIR	Survey of semantic-gap reduction, feature extraction, and similarity measures	Identified that retrieval quality depends on three foundations: discriminative visual features, suitable similarity metrics, and methods that reduce the semantic gap between low-level pixels and user intent.
2	Datta et al., 2008	Evolution of image retrieval	Large-scale survey of image retrieval methods, evaluation, and applications	Established CBIR as a major research area and emphasized the growing need for scalable, machine-learning-driven retrieval frameworks beyond keyword annotation.
3	Babenko et al., 2014	CNN descriptors for retrieval	Extraction of “neural codes” from deep CNN activations with PCA compression	Showed that pretrained CNN activations can serve as strong global image descriptors for retrieval, even when the original network was trained for classification rather than retrieval.
4	Wang et al., 2014	Fine-grained image similarity	Triplet-based deep ranking network with multiscale learning	Demonstrated that deep ranking learns similarity relationships more effectively than handcrafted descriptors and improves retrieval for visually close categories.
5	Redmon et al., 2016	Real-time object detection	YOLO unified single-stage detection framework	Proved that real-time object localization can be integrated into vision pipelines efficiently, which is valuable for ROI-guided search where background noise must be reduced before feature extraction.
6	Gordo et al., 2016/2017	End-to-end deep retrieval	Siamese/triplet deep architecture built on R-MAC for retrieval optimization	Reported major gains in instance-level image retrieval by learning retrieval-specific deep representations instead of reusing generic classification features alone.
7	Selvaraju et al., 2017	Explainable AI for CNNs	Gradient-weighted class activation mapping (Grad-CAM)	Introduced a generic visual explanation mechanism that highlights the image regions influencing CNN decisions, making deep retrieval systems more transparent and trustworthy.
8	Johnson et al., 2017	Billion-scale similarity search	GPU-accelerated nearest-neighbor search and k-selection for large vector sets	Demonstrated that high-dimensional similarity search can be performed at very large scale with major speedups, forming the algorithmic basis behind practical FAISS-style indexing.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

9	Sandler et al., 2018	Efficient deep backbone design	Inverted residuals and linear bottlenecks in MobileNetV2	Showed that MobileNetV2 achieves a strong accuracy-efficiency tradeoff, making it well suited for real-time or resource-aware visual search systems that require compact embeddings.
10	Radenović et al., 2019	Retrieval fine-tuning without manual labels	Automated training pair generation using 3D reconstruction and CNN fine-tuning	Proved that retrieval-focused CNN descriptors can be significantly improved without human annotation, reinforcing the importance of retrieval-specific optimization for image search systems.

III. BACKGROUND WORK

Visual similarity search is a specialized branch of content-based image retrieval in which the objective is to identify images that are visually close to a given query image without relying on textual tags or manual annotation. In conventional digital repositories, search is often performed using filenames, captions, or manually assigned keywords. Although such methods are easy to implement, they fail when the textual description is incomplete, inconsistent, or unable to reflect the actual visual structure of the image. This limitation becomes severe in large-scale repositories such as e-commerce catalogs, medical image collections, digital libraries, and wildlife image archives, where visual appearance is often more important than textual description. Your project addresses this problem by adopting a deep learning-based pipeline in which image understanding is performed directly from pixel-level content, followed by high-speed vector retrieval and explainable output generation.

At the foundation of modern visual search systems lies the idea of transforming an image into a compact numerical representation that preserves its semantic and structural information. These numerical representations, usually referred to as embeddings or feature vectors, serve as mathematical signatures of visual content. In your system, this role is performed by MobileNetV2, which converts the uploaded image or the selected region of interest into a dense feature vector. The vector is later compared with stored vectors from the image repository to retrieve the most visually similar matches. This shift from keyword-based retrieval to embedding-based retrieval is what makes the proposed framework more adaptive, scalable, and accurate for real-world multimedia applications.

A. Content-Based Image Retrieval

Content-Based Image Retrieval, commonly abbreviated as CBIR, refers to the process of searching images by analyzing the content of the image itself rather than depending on associated text. Traditional CBIR systems used handcrafted descriptors such as color histograms, texture descriptors, edge maps, and local feature operators. While these techniques offered reasonable performance in controlled environments, they often lacked semantic understanding. For example, two visually similar objects may not produce close matches if their handcrafted low-level features vary due to illumination changes, scale variation, background clutter, or viewpoint differences. This motivated the transition toward deep learning-based descriptors, which can automatically learn hierarchical feature representations from large image datasets. Your project fits into this modern CBIR paradigm by replacing handcrafted features with deep semantic embeddings derived from MobileNetV2.

B. Deep Feature Extraction

Feature extraction is the most critical component in a visual search system because retrieval quality depends heavily on how well the embedding represents the input image. Deep convolutional neural networks have become the preferred choice for this task because they learn progressively richer feature hierarchies, starting from edges and textures in early layers and progressing toward object parts and semantic patterns in deeper layers. In your project, MobileNetV2 is used as the feature extraction backbone because it provides a strong balance between representational power and computational efficiency. The document states that the system bypasses the final classification head and uses the global average pooled representation as a 1280-dimensional feature vector, which acts as a mathematical fingerprint of the image. This makes the model suitable for visual retrieval while remaining efficient enough for real-time use.

C. Region-of-Interest Selection and Object Isolation

A major challenge in visual similarity retrieval is that the image may contain irrelevant background information that reduces retrieval precision. If the system extracts features from the full image without isolating the important subject, then background textures, lighting artifacts, or unrelated surrounding objects may distort the embedding.

To overcome this issue, your project introduces YOLOv8-nano to detect candidate objects and allow region-of-interest selection before feature extraction. This improves the retrieval pipeline by focusing the embedding process on the visually meaningful part of the query image. As described in the system architecture and workflow, object isolation reduces noise and improves search precision, especially in domains where the target object occupies only a portion of the full image.

D. Similarity Matching and Vector Search

Once the feature vector is extracted, the next challenge is to compare it efficiently against a large repository of stored image vectors. A naive linear comparison becomes expensive when the image collection grows. Your project addresses this using FAISS, a vector similarity indexing framework designed for fast nearest-neighbor search. The document explains that cosine similarity is used as the matching criterion and that normalized vectors are stored in a FAISS index to achieve millisecond-level retrieval. This is a crucial design decision because it converts the retrieval problem into a high-dimensional vector search problem, enabling both speed and scalability. In practical terms, this means that the system can return top-K similar matches rapidly, even when the image repository becomes moderately large.

E. Explainable AI in Visual Search

One of the common criticisms of deep learning systems is that they operate as black boxes. In visual retrieval, users may receive similar images without understanding why those images were selected. To improve trust and interpretability, your project incorporates Grad-CAM-based explainable AI. This module highlights the image regions that contributed most strongly to the model's decision, allowing the user to understand which visual cues were prioritized during retrieval. The project document explicitly states that Grad-CAM heatmaps are generated from deep feature activations and overlaid onto the query or matched images. This makes the system not only functional but also transparent, which is particularly important for sensitive applications such as medical imagery, evidence analysis, and decision-support systems.

F. Real-Time and Scalable Retrieval

Modern intelligent multimedia systems must not only be accurate but also responsive and scalable. The uploaded report repeatedly emphasizes that the project is designed for real-time performance, using a lightweight MobileNetV2 backbone, efficient FAISS search, and a Flask-based web interface. It also mentions future expansion toward cloud deployment, mobile search, and larger datasets. This indicates that the system is built with modularity and scalability in mind. Therefore, the background work of your project is not limited to image retrieval theory alone; it also involves efficient web integration, asynchronous handling of user requests, vector storage, and explainable output delivery.

IV. PROPOSED MODEL

The proposed model presents an integrated visual similarity search framework that combines deep learning, object detection, vector indexing, and explainable AI into a unified retrieval pipeline. The main objective of the model is to accept a user-provided image, optionally isolate the most relevant object region, convert the selected visual content into a compact feature embedding, compare that embedding against a database of stored embeddings, and return the most visually similar results along with supporting explanations. This design directly addresses the shortcomings of traditional keyword-based retrieval, particularly the dependence on manual annotations and the lack of semantic understanding.

A. Model Overview

The proposed system is designed as a multi-stage pipeline. It begins with a query image uploaded through the web interface. The uploaded image then passes through preprocessing operations such as resizing and normalization. If required, YOLOv8-nano performs object detection so that the system can isolate a meaningful region of interest. The selected image or ROI is then fed into MobileNetV2, which extracts a deep semantic feature vector. This vector is normalized and compared with previously stored feature vectors in a FAISS index using cosine similarity.

The top-ranked matching images are returned to the user through the interface, and Grad-CAM is employed to provide heatmap-based explanations of the model's retrieval behavior. This architecture is directly reflected in the system workflow and architectural diagram in the document.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

B. Image Input and Query Handling

The first module of the proposed model is responsible for accepting user input. The web platform allows users to upload images in standard formats such as JPG and PNG. This uploaded image becomes the query against which the system performs similarity retrieval. A well-designed query input stage is important because it defines the quality and usability of the retrieval system. The interface is intended to be simple and responsive, allowing even non-technical users to initiate search tasks easily. The project also includes secure login and session management in its broader system functionality, which indicates that the query flow is part of a structured application environment rather than a standalone script.

C. Preprocessing and Standardization

Before feature extraction, the input image must be standardized so that it is compatible with the deep learning model. In this stage, resizing, normalization, and format conversion are applied. These steps are essential because CNN models expect consistent input dimensions and normalized pixel distributions. Without standard preprocessing, extracted features may become unstable and retrieval performance may degrade. The system document states that preprocessing is used to improve feature extraction accuracy and ensure that the image conforms to the input requirements of MobileNetV2. This preprocessing module therefore acts as the bridge between raw image input and learned semantic representation.

D. ROI Detection Using YOLOv8-nano

A distinctive aspect of the proposed model is the integration of YOLOv8-nano for object detection and optional ROI cropping. Instead of always analyzing the full image, the system can focus on a specific object chosen from the detected bounding boxes. This is especially useful when the image contains cluttered backgrounds or multiple objects. By isolating the target region before embedding generation, the system improves search specificity and reduces the influence of unrelated visual elements. The report clearly notes that YOLOv8 is used to perform zero-shot or real-time object detection and ROI bounding box generation, making this stage an important accuracy-enhancement module in the retrieval pipeline.

E. Deep Feature Extraction Using MobileNetV2

After preprocessing and optional ROI selection, the image is passed through MobileNetV2. This network is used not as a classifier but as a feature encoder. The final classification head is excluded, and the intermediate or pooled representation is taken as the image embedding. According to the project document, the output is a 1280-dimensional feature vector that captures visual structures such as textures, edges, and semantic patterns. The choice of MobileNetV2 is significant because it offers reduced computational overhead while maintaining discriminative capability. This makes it highly suitable for systems that must run efficiently in real time and possibly in CPU-friendly environments.

F. Vector Storage and FAISS Indexing

Once feature embeddings are extracted, they must be stored and indexed efficiently. The proposed system uses FAISS to manage this high-dimensional vector repository. Each image in the dataset is represented by a vector, and these vectors are inserted into the FAISS index. This allows fast nearest-neighbor retrieval when a new query vector is provided. The document mentions that dataset vectors, metadata, and FAISS binaries are managed as part of the storage layer, showing that the project treats indexing as a central system component rather than a simple add-on. This design is essential for scalability because brute-force comparison becomes increasingly expensive as the repository grows.

G. Similarity Matching and Ranking

After query vector generation, the retrieval engine compares it with stored vectors using cosine similarity. Cosine similarity is well suited to normalized feature vectors because it measures angular closeness rather than raw magnitude difference. The top nearest neighbor are then ranked according to similarity score and presented to the user. This ranking strategy is important because it turns raw similarity computation into a meaningful search output. The system workflow described in the report specifically notes that the most similar images are retrieved and displayed in ranked order, enabling effective visual browsing and analysis.

H. Explainable Visualization with Grad-CAM

To make the retrieval process understandable, the system applies Grad-CAM to generate visual heatmaps over the query and matched images. This allows the user to inspect which regions influenced the embedding or similarity decision most strongly.

Such interpretability is valuable in research and practical deployments because it helps validate that the model is focusing on semantically relevant content rather than irrelevant artifacts. The uploaded project document describes this explicitly as a means of solving the black-box problem and improving trust.

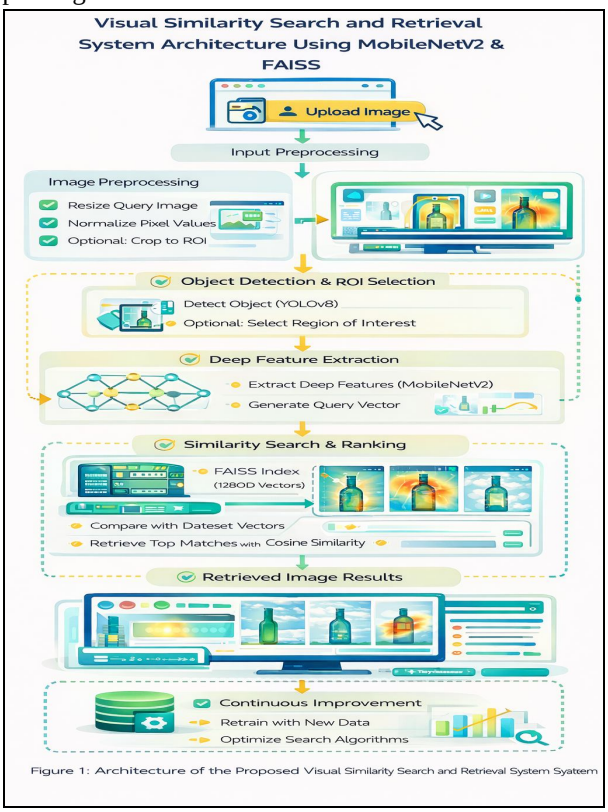


Figure 1: Architecture of the Proposed Visual Similarity Search and Retrieval System Using MobileNetV2 and FAISS

Figure 1 illustrates the overall architecture of the proposed visual similarity search system. The process begins with image upload, followed by preprocessing steps such as resizing, normalization, and optional region-of-interest (ROI) selection using YOLO-based object detection. The processed image is then passed to the MobileNetV2 deep learning model for feature extraction, where high-level visual features are converted into numerical vectors. These feature vectors are stored and compared using FAISS-based indexing, enabling efficient similarity search. Cosine similarity is applied to retrieve the most relevant images from the dataset. The retrieved results are displayed through an interactive interface, and the system is further enhanced through continuous improvement mechanisms such as retraining and optimization.

I. Output Interface and User Feedback

The final results are displayed through the web interface. The output includes ranked matching images, visual similarity indicators, and explainable AI heatmaps. The report also references GSAP-based animations and a premium frontend layout, suggesting that user experience is treated as an important part of the system rather than an afterthought. This output layer ensures that the retrieval results are both informative and visually interpretable, which strengthens the practical usability of the model.

V. IMPLEMENTATION RESULTS

The implementation of the proposed visual similarity search system demonstrates how deep learning and vector search can be integrated into a functional web-based retrieval platform.

The implementation is divided into multiple modules including user interaction, preprocessing, feature extraction, vector indexing, similarity matching, and visualization. These modules operate together to ensure that the system delivers accurate and fast retrieval results for uploaded query images. The report's implementation section confirms that the system was structured modularly to remain scalable, efficient, and easy to maintain.

1) Login Page

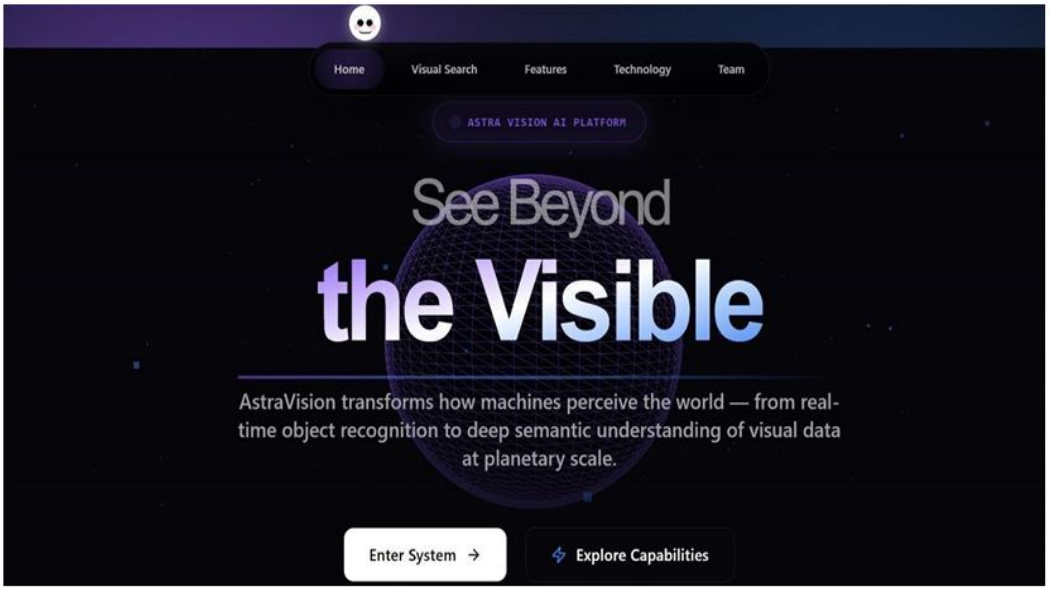


Figure 2: Login Page - Enter Authorization ID and Passkey

Figure 2 explains about the initial access point of the system that allows authorized users to securely enter the application. This interface typically contains input fields for the username and password along with a login button to authenticate the user. It ensures that only valid users can access the emotion detection system and its features.

2) Image Upload for Object Detection Dashboard

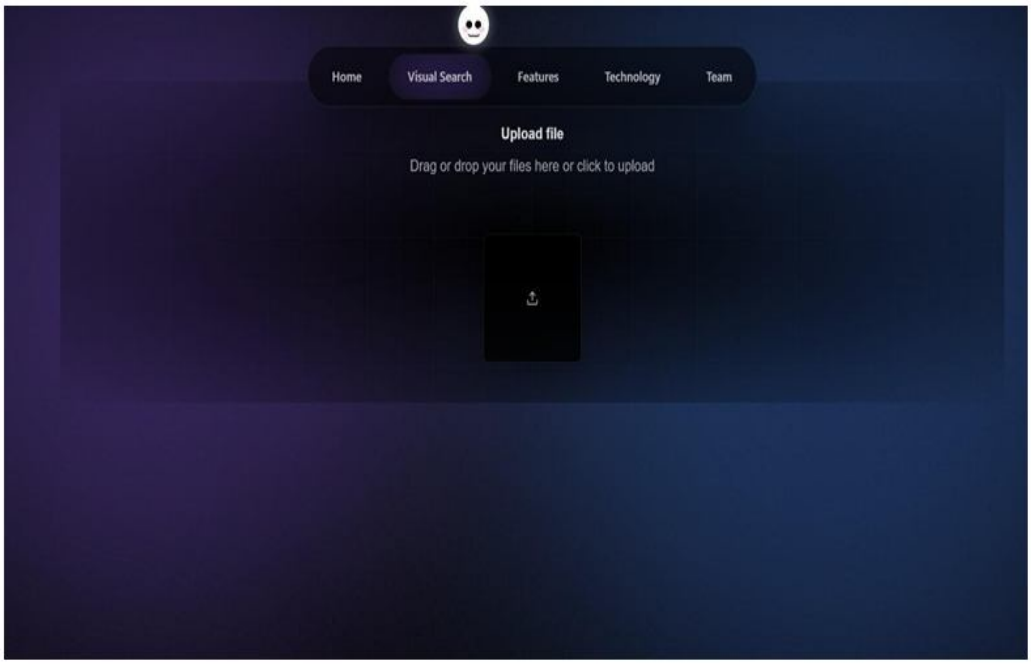


Figure 3. YOLOv8-Based Object Detection in Query Image

Figure 3 explains the interface where the system takes user input and try to process that image using YOLOv8 model and then check which object is matched with input image.

3) Results Page

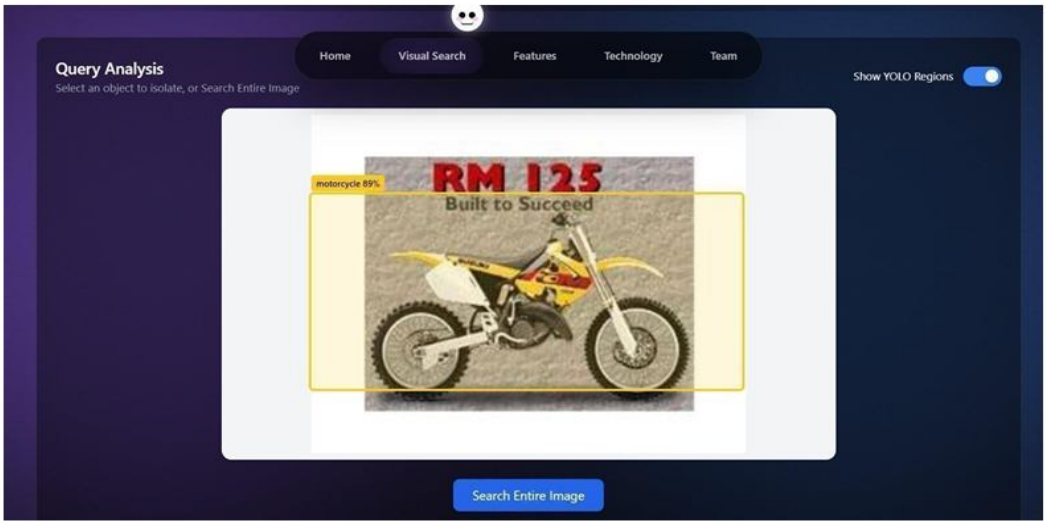


Figure 4. Visual Similarity Search

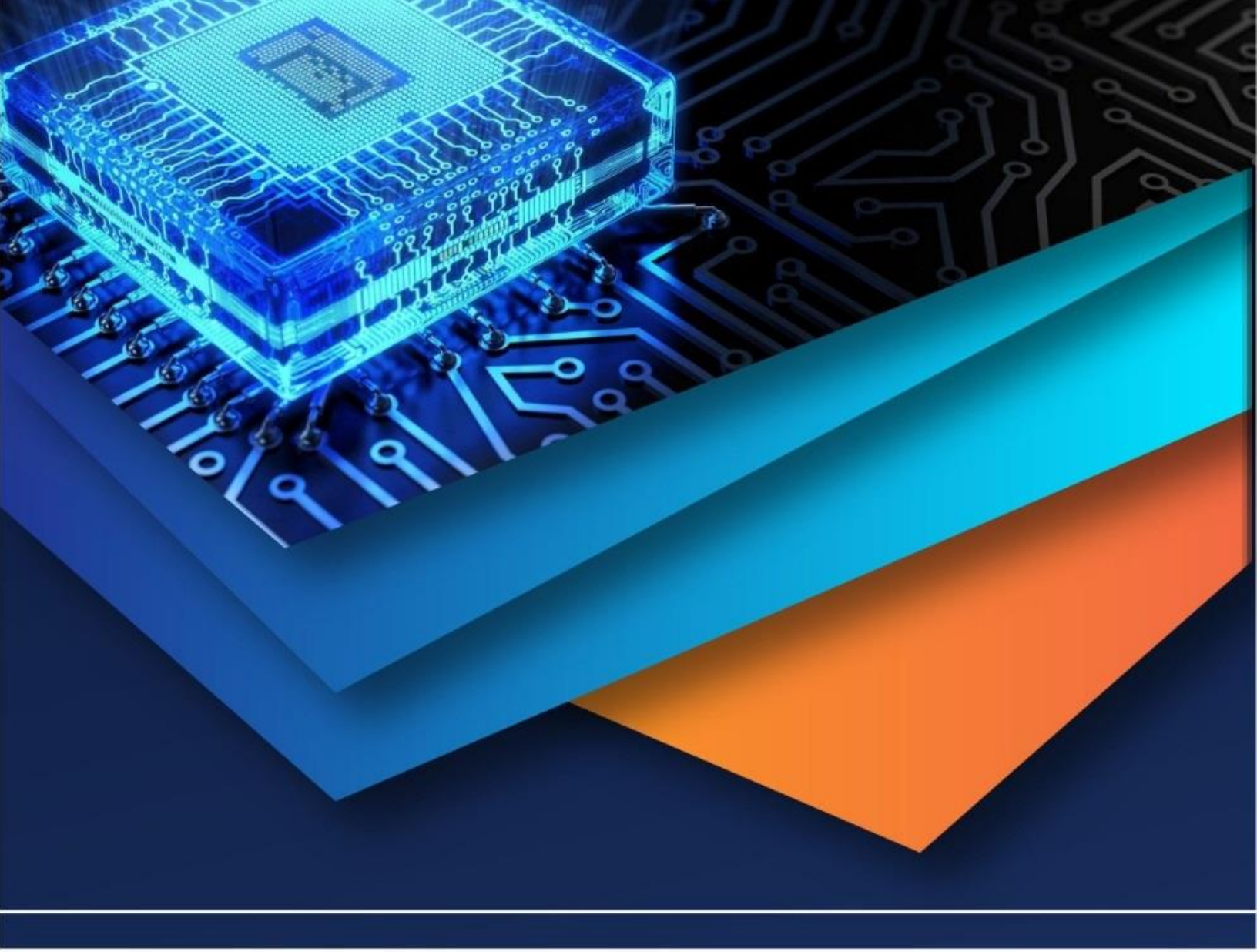
Figure 4 shows the output of the proposed application based on visual similarity search.

VI. CONCLUSION

This research presents an efficient and scalable real-time facial affect recognition system that leverages deep learning and computer vision techniques for accurate emotion detection. By utilizing the VGG-Face architecture for feature extraction, the system is capable of capturing complex facial patterns and translating them into meaningful emotional representations. The integration of OpenCV-based preprocessing ensures reliable face detection and normalization, enabling consistent performance under varying environmental conditions. The proposed framework successfully combines a robust backend with an interactive web-based interface, allowing users to perform emotion analysis through webcam input or image uploads. Despite its effectiveness, the system has certain limitations such as sensitivity to extreme lighting conditions and challenges in detecting subtle expressions. Future work can focus on improving accuracy using larger and more diverse datasets, incorporating multi-face detection, and extending the system to multimodal emotion recognition by integrating audio and physiological signals. Additionally, deployment on mobile and cloud platforms can further enhance accessibility and scalability. In conclusion, the proposed system demonstrates the practical applicability of deep learning in emotion-aware intelligent systems and contributes toward advancing human-computer interaction by enabling machines to understand and respond to human emotions more effectively.

REFERENCES

[1] Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma, “A survey of content-based image retrieval with high-level semantics,” Pattern Recognition, vol. 40, no. 1, pp. 262–282, 2007.
[2] R. Datta, D. Joshi, J. Li, and J. Z. Wang, “Image retrieval: Ideas, influences, and trends of the new age,” ACM Computing Surveys, vol. 40, no. 2, article 5, 2008.
[3] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky, “Neural codes for image retrieval,” in Proc. European Conference on Computer Vision (ECCV), 2014, pp. 584–599.
[4] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, “Learning fine-grained image similarity with deep ranking,” in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 1386–1393.
[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788.
[6] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, “End-to-end learning of deep visual representations for image retrieval,” International Journal of Computer Vision, 2017/2018.
[7] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual explanations from deep networks via gradient-based localization,” in Proc. IEEE International Conference on Computer Vision (ICCV), 2017, pp. 618–626.
[8] J. Johnson, M. Douze, and H. Jégou, “Billion-scale similarity search with GPUs,” IEEE Transactions on Big Data / preprint version available from 2017.
[9] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 4510–4520.
[10] F. Radenović, G. Tolias, and O. Chum, “Fine-tuning CNN image retrieval with no human annotation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 41, no. 7, pp. 1655–1668, 2019



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)