OPEN ACCESS

Kashf Journal of Multidisciplinary Research
Vol: 03 - Issue 03 (2026)
P-ISSN: 3007-1992 E-ISSN: 3007-200X
https://kjmr.com.pk

ADVANCED FEATURE REPRESENTATION IN CONTENT-BASED IMAGE RETRIEVAL UTILIZING DEEP LEARNING FRAMEWORKS

Muhammad Mohsin[1], Jahan Khan[2], Muhammad Faisal[3], Asim Abdul Qadir[4], *Muhammad Arslan[5], Sohail Raza Chohan[6]

[1, 2, 3, 5, 6]Department of Computing & Emerging Technologies, Emerson University, Multan, Pakistan.

[4]Department of Software Engineering, National University of Modern Languages, Multan, Pakistan.

***Corresponding Author:** (arslan.shabbir@eum.edu.pk)*

DOI:(https://doi.org/10.71146/kjmr880)

Article Info

Abstract

Content-Based Image Retrieval (CBIR) systems aid in the retrieval of images by utilizing the inherent visual attributes as opposed to the manual textual labels, providing a better alternative to the conventional metadata-based search. With the growing rate of development of digital repositories in key domains such as medical imaging, surveillance, and digital forensics, the need to undertake automated, scalable and precise image analysis has been stressed. But modern CBIR systems are often faced with bottlenecks including poor feature extraction, ineffective preprocessing pipelines, and insensitivity to environmental changes such as changing lighting and background noise. To overcome these shortcomings, this research will present a new improved CBIR architecture with streamlined image-processing pipeline. The suggested methodology will shift the input of colors to the regular grayscale form since the calculated scale is normalized to reduce the computational burden without affecting the integrity of the structure. A strict preprocessing pipeline including Contrast Limited Adaptive Histogram Equalization (CLAHE), spatial normalization and data augmentation introduced to guarantee feature consistency and resistant Ness to noise. A Convolutional Neural Network (CNN) provides the feature representation, to be more precise the state of the art MobileNetV2 design is used, which employs depth wise separable convolutions to extract features high-effectively. Extraction of feature vectors is handled through an effective indexing scheme to do fast similarity matching. Experimental evidence shows that the proposed system is much better than the currently available baseline approaches in retrieval accuracy (98.06/97.04) and computational efficiency. The results affirm that the system is robust and can be deployed in real-life and high-volume image retrieval systems.

Keywords:

Content-Based Image Retrieval, Convolutional Neural Networks, MobileNetV2, Feature Extraction, CLAHE, Retrieval Accuracy.

1. INTRODUCTION

The accelerated growth of digital technology, the spread of devices linked to the Internet and the growth of cloud computing have left altogether to an unprecedented and exponentially large increase in data output in multimedia platforms[1]. In this growing digital universe, unstructured data especially high-resolution digital images, takes up colossal and ever-increasing amount of stored information. In such critical, data-intensive industries as healthcare diagnostics, security and surveillance, digital forensics, and e-commerce millions of visual assets are recorded, uploaded, and shared daily. With the volume, velocity, and types of these image libraries ever growing in size at an appalling pace, the need to have strong, automated and highly accurate retrieval mechanisms has moved beyond being a hypothetical computer science problem to becoming a pressing, real-world demand. This presents significant challenges in terms of navigating these huge bodies of visual data that necessitate effective searches to extract practical information, make patterns, and make fast decisions in fast-paced settings where time and precision are paramount to success[2].

Text-Based Image Retrieval (TBIR)[3] has traditionally been used as the basis of the methodology on how to manage and search these developing visual repositories. TBIR is metadata-driven and appears to be more manual-based with annotations and keywords and descriptions to classify and index images[4]. In the retrieval stage, a textual query by a consumer is searched and compared to these annotations, which have been previously built, to retrieve the pertinent results. Though the first TBIR systems had a lot to offer in terms of organizing digital content, human subjectivity and physical constraints in a fundamental manner limit these systems[5]. Hand tagging is so consuming that it is very non-scalable in the current era of Big Data. Also, there are language differences and disparities in vocabulary and culture that plague the TBIR systems. Above all, TBIR often does not fill in the so-called semantic gap the natural and complex mismatch between high-level textual instructions given by a human and the low-level, intrinsic visual information distributed in the pixels of the image[6].

Content-Based Image Retrieval (CBIR)[7][8] systems have become an important hotspot in contemporary information retrieval literature as a way to counteract the severe shortcomings of manual tagging and metadata dependency [6]. The automated and highly scalable option projected by CBIR, is based on the fact that the retrieval paradigm needs to be changed to focus not on external and human-created metadata, but on the intrinsic characteristics of the image itself. A CBIR system will extract and experiment on intrinsic visual features (shape boundaries, color distribution, texture patterns) flesh directly out of the query image[9] and not do it by using arbitrary tags. On entering a query image, the system transforms the image into a multi-dimensional feature space mathematically, isolating the important feature vectors. Such extracted vectors are then systematically matched with the feature vectors of the images already stored in the database, which were computed beforehand. All of this mathematical comparison often involves distance measures to compute visual similarity, including the Euclidean distance formula[10] by automating evaluation of visual data, CBIR[8] systems provide a more objective and scalable method of managing multimedia.

Although this automated method has structural and theoretical advantages, more often than not, conventional CBIR models experience major bottlenecks in their operations. Detecting extremely precise matches in a variety of and large-scale databases is a difficult matter as there are environment differences inherent in any real image. Both environment luminance variations, extremely complicated or cluttered backgrounds, occlusion and noise created by sensors have significant effects on the quality of features extracted[11]. The feature extraction algorithms (e.g., SIFT, HOG) commonly used as traditionally, handcrafted algorithms have a hard time accommodating these variations, which severely constrains the overall system retrieval accuracy, precision and environmental resilience. As a result, a solidly working retrieval system with a controlled, lab based simple environment can completely stop working with uncropped and uncontrolled images in the live database[12].

The incorporation of deep learning paradigms into the CBIR systems will be a ground-breaking motion in order to address these computational and environmental problems. This paper hypothesizes a refined CBIR architecture as a systematic way of reducing the conventional bottlenecks by performing an extensive standardized preprocessing workflow before the essential feature extraction phase[13]. In particular, the full-fledged image enhancement algorithms as the Contrast Limited Adaptive Histogram Equalization (CLAHE) have been used. The process of CLAHE[14] optimally takes advantage of the contrast of the region of interest and inhibits mathematically the amplification of the background noise so that the representation of features in CLAHE can be constantly the same with reference to the lighting conditions when the picture was captured[15]. The system can eliminate computation space and structural integrity through the transformation of inputs into a normalized scale which is deemed to bypass the heavy computational load whilst preserving the structural integrity, forming a clean and standardized base on which the neural network can operate[16].

After this ideal phase of preprocess, the use of Convolutional neural networks (CNNs) is used to completely substitute conventional, manual, extraction mechanisms. CNNs have a strong hierarchical learning capability; their hierarchical convolutional and pooling operations and non-linear activation functions automatically compute the transformation of raw and complex visual representations to highly discriminant feature vectors[17]. Compared to manual extraction, deep learning models are able to record the complex spatial hierarchies in an image by themselves[18]. This study seeks to deliver a really strong indexing mechanism by examining the strategic synergising of optimised image preprocessing and deep CNN-based feature extractions. Moreover, these sophisticated architectures need to be effectively incorporated with effective indexing mechanisms, which can facilitate quick similarity matching to be used in lightweight, easy to use web applications[19]. This not only means data that the system outperforms in terms of precision, recall, and computational efficiency base line methods significantly, but allows scaling of a accessible system to end-users observing visual data in large volumes

2. LITERATURE REVIEW

The Content Based Image retrieval (CBIR)[8] has seen a dramatic change whereby what used to be mere color and texture matching to something that is intelligent in nature and structure[20]. This part represents a synthesis of new developments within the latest field, namely the movement towards hybrid feature fusion and deep learning-based models. Through reviewing and evaluation of various methodologies including but not limited to bi-layer search strategies to accuracy noise reduction, this review recognizes the manner in which modern systems bridges the so-called semantic gap. Their performance on benchmark datasets such as Corel-1K is given special attention, but it can be seen that an overall upward trend has been observed in terms of accuracy and computation throughput. In turn, the next synthesis forms the technical basis of the offered multi-layered approach elaborated within the current work.

Vieira et al. [21] introduce a software product called CBIR-ANR, that addresses the demands of an explosion of multimedia information of smartphones and social networks. Their main contribution consists of an "Accuracy Noise Reduction" (ANR) approach which serves as a post-processing mechanism to modify query responses. Using this strategy, the system enhances an image retrieval process towards assertiveness by eliminating an irrelevant visual noise that usually results in false positive. Their approach combines three low-level features that are concatenated, to present a small feature of 187 dimensions. The reason behind this particular choice of vector size is to provide a compromise between the efficiency of computation and the descriptive capability of a particular model, and as such, is very appropriate when dealing with large-scale data in which processing time and storage are vital considerations[22]. The authors show that such simplified feature representation, combined with the ANR strategy are competitive to the more complex state-of-the-art models. This is the main advantage

to the strategy, as it is capable of fine-tuning retrieval performance without needing the computational scales of deep learning models, and is a powerful way to help individuals, governments and businesses search through large volumes of image data much more accurately and with lower memory consumption.

Salih and Abdulla [23] introduce a hybrid, bi-layer Content-Based Image Retrieval method, which is more focused on retrieval performance rather than computation performance. They use a methodology that splits the search process into two levels of filtering to be as precise as possible. A Bag of Features (BoF) method with speed-up robust features (SURF) is used to match the query image with the whole database in the first layer, and to eliminate as many non-similar images as possible to reduce the search space[24]. The second layer then narrows down the remaining candidate images by generating global features, namely focusing on texture (through Local Binary Pattern and Discrete Wavelet Transform) and color (through RGB, HSV and YCbCr spaces). The system compares each single color channel obtaining the entropy and the mean, bringing a highly granular comparison. The effectiveness of this hierarchical approach was confirmed through the experimental results carried out with the benchmark dataset of the Corel-1K. The results showed that the bi-layer technique achieved the highest top-10 precision rate of 86.65 and top 20 precision of 81. This paper highlights the importance of integrating local interest points to pre-filter them with exhaustive global feature extraction to refine it and boost the refined semantic gap as well as outperform traditional single-layer information retrieval techniques.

Wangi and Makandar [25] investigate the potential of Transfer learning relating to Content-Based Image Retrieval[8] using pre-trained answers of the Convolutional Neural Network (CNN). Their study aims at countering the shortcomings on handcrafted characteristics by exploiting the hierarchical representations that have been taught by architectures such as VGG16 and ResNet50. The methodology proposed is a complex feature fusion approach which incorporates various visual features observed at the different layers of these deep networks. When combined, the representations on these multi-levels result in a more detailed and robust image content description than any single layer or conventional feature would have offered. The paper underlines that the concept of deep learning models which have been trained on large datasets already has an intricate set of visual shapes, textures, which can be successfully reused in particular seeking tasks. Experiments over standard benchmark datasets proved that this CNN-based fusion methodology adds better retrieval and makes it more robust. The findings underscore the capabilities of deep transfer learning to take the state of art in CBIR to the next level to offer an arguably more reliable tool to the multimedia contents management and the analysis of complex images where the conventional, rigidly structured saw models fail to view the image as per the underlying semantic meaning[26].

Fawad and his coworker [27] examine the effectiveness of the deep learning-based proposal methodologies to develop a more effective image retrieval system. Their relative analysis compares three different architectures: a regular Convolutional Neural Network (CNN), a hybrid network comprising of convolutional layers merged with Long Short-Term Memory (LSTM), and a hybrid network containing convolutional layers and Gated Recurrent Units (GRU). Combining recurrent (LSTM and GRU) NN layers with spatial CNN layers, the authors sought to capture finer patterns in the visual data. These models were strictly evaluated on four different benchmark databases of different size: Corel-1K, Cifar-10, Cifar-100, and Mnist 70K[28]. The experimental results suggested that the hybrid models decreased the computation time significantly and offered high standards of accuracy. In the case of the Corel-1K dataset, the CNN-GRU model got the best result with the accuracy of 95.5%, the CNN-LSTM and the basic CNN models came second with 94.5 and 93.3, respectively. The CNN-GRU architecture also achieved an astonishing 87.5% accuracy even in the more difficult Cifar-100 dataset. These results demonstrate that the combination of spatial feature extraction and gated recurrent networks is a more dynamic and intelligent retrieval architecture that can easily navigate both large-scale and complex visual archives with minimum processing delays.

Ghaleb, and the research team [29] introduce an adapted Deep Convolutional Neural Network architecture called M-VGG16 which addresses the difficulties of large repositories of images. Their method is a structural change of the classic VGG16 network by introducing specialised convolution kernels in the input layer, to be followed by the depth concatenation of the outputs. To make the obtained feature vectors robust and computationally efficient, they use Principal Component Analysis (PCA) to illustrate dimensionality in the obtained features. This uncovers a smaller index of image features in which the dimension of the feature vectors is minimized by 88 percent than the starting M-VGG16 framework. This M-VGG16 + PCA architecture was tested on four test datasets Corel-1K, Corel-10K, Coil-10K, and KADID-10K. The findings showed that the proposed model always performed better than state-of-the-art models such as AlexNet and regular VGG16 in terms of precision, recall, and F1-score. Through the addition of modified deep feature extraction and an effective lineal dimensionality reduction the authors have been able to create a system that provides high quality retrieval results whilst reducing the overall computer and storage needs significantly and this means that the system is highly useful both in medical repositories of images and in the overall web based multimedia content[30].

Kumar and Singh [31] are concerned about resolving the discrepancy between human perception and machine-based extraction by using Deep Adaptive Attention Network (DAAN). Appreciating the fact that local classifier systems of traditional CBIR traditionally use rigid, hand-created descriptors, their DAAN architecture adopts transformer-driven models in combination with deep neural networks (DNN) to obtain spatial representations and contextual relationships of an image. The central aspect of their approach is the Adaptive Multi-level Attention (AMLA)[32] module that dynamically guarantees correct weighting of the features. This enables the system to concentrate in the minute visual alterations that are semantically significant but do not touch on the unimportant background information. Their study results indicate that compared to current methods, the DAAN-CBIR framework has an improved Mean Average Precision (mAP) and retrieval speed in addition to the training time is much shorter. The researchers suggest that this adaptive attention process is critical to up-to-date applications such as e-commerce and individual media suggestions as well as preservation of digital information. The authors have applied the combination of CNNs, with their ability to extract space features, and transformers, with their capacity to understand situations, to build a highly powerful and precise system capable of accommodating the particular challenges of diverse datasets in many professional fields[32].

Choe et al. [33] investigated a niche of Content-Based Image Retrieval; that of clinical diagnostics, namely detection of Interstitial Lung Diseases (ILD) in CT images of the chest. Since the task of ILD assessment is also quite complicated, and inter-reader variability is quite high so that the researchers decided to create a deep learning-based C]

IR system to be used as a decision-support tool. The algorithm measures the spread and intensity of localized patterns of diseases, including Usual Interstitial Pneumonia (UIP) and Nonspecific Interstitial Pneumonia (NSIP), and recalls the top three closest images in an authenticated database of confirmed past instances. A clinical trial of eight readers of different experience levels revealed that the CBIR system do not only significantly enhance the accuracy of diagnostic results, but also gave less-experienced readers tangible visual cues, which enhances diagnostic accuracy. Results of the experiment indicated that the system had a pattern-based accuracy of 73.4% and top-1 accuracy of 77.7% with several regions of interest analyzed. The study shows that pattern-based retrieval may normalize diagnostic confidence in a very complicated medical domain and may be able to give the critical contextual clues that would be required to fill the gap between the low-level data on pixels and high-level medical diagnosis, potentially leading to better rates of errors in time-sensitive clinical settings.

Fernandes and associates [34] discuss the medical annotation inefficiency when it comes to capsule endoscopy video images (VCE). Now, these videos have to be manually annotated by specialists, which is infamously time-

intensive, costly and open to human error. To address these problems, the authors suggest the Deep Learning and CBIR-based solution based on the use of a ResNet-18 architecture based on the Siamese network. It is a network that is trained to match medical images in pairs of images based on their features and determine if they are a match or not. It was tested on a large dataset of 5792 pairs of images and compared the different learning rates and optimizers such as Adam, SGD and Adadelta. The experimental results showed that the Adam optimizer gave the most promising results with an impressive accuracy of 97.6 and an Area under the curve (AUC) of 0.9764. This great degree of precision is evidence of the fact that the model can substantially save the amount of time spent by medical specialists on manual annotation. This Siamese-based CBIR network can significantly reduce the costs and enhance the reliability of VCE-based medical diagnosis by automating the search of identical frames in ameliorated or artificed video feeds thus also producing an accurate and efficient gastrointestinal examination tool.

Arulmozhi and Gopi point out [35] that searching by text is an easy and widespread method, whereas searching by visual content in terms of Content-Based Image Retrieval is much more concerned and better appropriate with current image databases. Their study is aimed at developing a smart CBIR algorithm enhancing the ability to retrieve information based on automated feature extraction and matching. Their methodology emphasizes the fact that an effective CBIR system should consist of two key stages: effective feature extraction to reduce the size of the required data to describe an image and accurate feature matching to rank the indexed images based on the distance between themselves and the query.

With help of the extracted features of the query image, the system can rank the findings based on the visual similarity instead of the metadata of the results by comparing the extracted features of the query image to the database using the specific distance measures. The authors believe that these kinds of intelligent systems are required in a wide range of disciplines such as weather forecasting, robotics and biomedical imaging. Their contribution promotes the incorporation of sophisticated methods of extraction that is able to reduce the essence of an image to a searchable form hence even with databases scaling to the millions of the search of useful visual data can be done both reliably and in a manner that is computationally viable to the application of the same, in real time.

Table 1: Comparison with previous work

Author(s)	Core Technique / Algorithm	Dataset	Key Result / Accuracy
Vieira et al. [21]	CBIR-ANR (Accuracy Noise Reduction) + 187D Vector	Large-scale sets	High assertiveness; size-efficient
Salih & Abdulla [23]	Bi-layer Search: SURF/BoF followed by Global LBP/DWT	Corel-1K	Top-10 Precision: 86.65%
Wangi & Makandar [25]	CNN Feature Fusion: VGG16 + ResNet50	Benchmarks	Superior accuracy via transfer learning
Fawad et al. [27]	CNN + GRU / LSTM Hybrid Architectures	Corel-1K	CNN-GRU Accuracy: 95.5%
Ghaleb et al. [29]	M-VGG16 + PCA (88% reduction in vector size)	Corel-10K	Robust features; reduced dimension
Kumar & Singh [31]	DAAN: Deep Adaptive Attention + Transformers	Varied Sets	High mAP; reduced training time
Choe et al. [33]	Deep Learning CBIR (ILD Diagnosis support)	80 CT queries	Top-1 Accuracy: 77.7%

Fernandes et al. [34]	Siamese Network (ResNet-18) for Endoscopy	5,792 pairs	Accuracy: 97.6%
Arulmozhi & Gopi [35]	Intelligent Feature Matching	Various	Enhanced sensitivity vs. text search

3. METHODOLOGY

The session introduces the specifics of the methodology of the proposed deep learning-based Content-Based Image Retrieval (CBIR) system. The system combines Convolutional Neural Network (CNN) classification and feature-based image matching to have very high semantic matching.

The system uses a two-stage architecture, in which a supervised classification step is used to overcome the semantic gap, followed by extracting deep feature embeddings in a dense intermediate layer. This methodology can be systematically separated into the preparation of the datasets, sophisticated preprocessing, the design of the CNN architecture, training plan, deep feature extraction, similarity, and performance assessment.

3.1 DATA ACQUISITION AND PREPARATION.

Any strong deep learning model is based on the content and structure of its data. The data used in this CBIR system is based completely on the format of the commonly used CIFAR-10 dataset.

- **Hierarchical Organization:** The data is organized into basic train and test directories. These directories also have subfolders that are subclassed according to the various object classes. This is to make sure that none of the images can fall into more than one of the ten semantic categories that have been defined.
- **Resolution and Storage Dualism:** Standardization to 32 x 32 pixels is used to store and archive the images. The given dimension is the optimal compromise that makes the images computationally efficient to experiment with fast and still captures the crucial visual and structural features needed in the process of deep learning.
- **Programmatic Loading:** Dynamically images are loaded by walking through the directory paths. In this stage, the system will store two versions of each image corresponding to grayscale and the original RGB image. The grayscale tensors get input into the CNN and are trained to enable the complexity of the computations to be reduced and the high fidelity RGB images are kept only to be displayed at its retrieval stage.

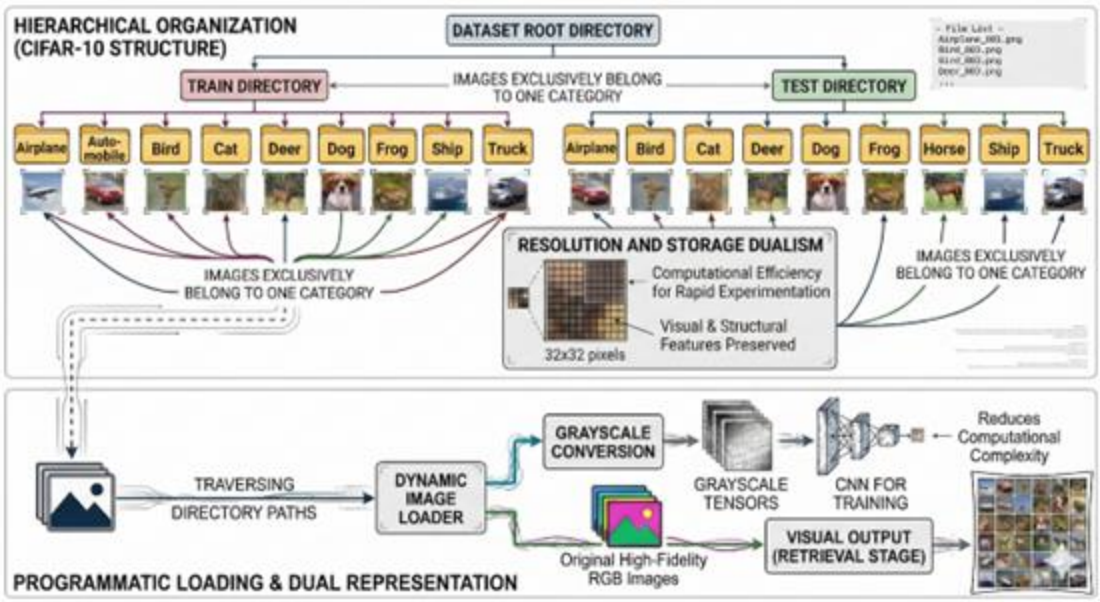

Figure 1: Data set acquisition and preparation

This flow chart represents the entire data structure and information processing workflow of an image retrieval system when using the CIFAR-10 data set. The upper segment is the hierarchical contacts of a folder tree with the root folder that is strongly separated into Train and Test folders that are further categorized into particular sub folders of each separate image category (e.g. an airplane, a cat, a ship). The lower part reflects the programmatic loading stage, in which the system dynamically loads these 32x32 pixel images in the directories and decomposes them into a "dual representation" workflow. To do all the mathematical hard work, the images are transformed into grayscale tensors and directly fed into a Convolutional Neural Network (CNN) to save computing resources in the training process. At the same time, the original full color RGB images are saved and fed directly into the final visual output stage, so when the system accesses similar images the images can be viewed in the original high-fidelity color.

3.2 ADVANCED IMAGE PREPROCESSING PIPELINE.

Image preprocessing is a non-negotiable and critical part of making models more stable, resulting in faster gradient descent, and converging in general. A set of stringent transformations are followed on the image by the pipeline.

3.2.1 GRAYSCALE CONVERSION

A process of grayscale conversion is used to lessen the dimensionality of the input images in terms of channels to one. This preserves the most important structural, geometric and texture details and leaves out the unnecessary color data. Weighted luminance transformation is used:

$$I_{Gray} = 0.299R + 0.587G + 0.114B \quad (1)$$

The conversion equation is of the form; where, is the final calculated Gray image intensity of the new grayscale pixel. This last value depends on R, G and B which represent the separate intensity values of one of the color channels used by each pixel in the original (i.e. Red, Green, and Blue) and each channel usually has a value between 0 and 255.

3.2.2 ENHANCING CONTRAST THROUGH CLAHE.

Contrast Limited Adaptive Histogram Equalization (CLAHE) is applied after the grayscale conversion to even out the lighting conditions. Common global histogram equalization methods tend to blur images and enhance background lisations. CLAHE, on the other hand, is based on local 8x8 pixel tiles (grids) and asserts a contrast threshold (at 2.0) to avoid noise enhancement in homogeneous areas.

3.2.3 PIXEL NORMALIZATION

Lastly, the pixel intensity values are normalized in order to make the gradient based optimization of the neural network numerically stable. Children values are adjusted to a more accurate range [0, 1] of 0-255:

$$I_{norm} = \frac{I_{Gray}}{255} \quad (2)$$

- $\mathbf{I_{norm}}$**:** The final "normalized" pixel intensity.
- $\mathbf{I_{Gray}}$**:** Your initial grayscale pixel value (between 0 (or all the way black) and 255 (or all the way white)).
- **255**: The brightest possible value in a typical 8-bit image.

3.4 MORPHEME REPRESENTATION

The categorical representations, which were extracted as strings in form of folder names, need to be mathematically altered that applies to the supervised learning stage.

- **Integer Mapping:** The LabelEncoder accepts any unique textual level and gives it a discrete integer level.
- **One-Hot Encoding:** These numbers are then turned into one-hot encoded vectors. In the case of a ten-class problem, the target vector is an array of 10 dimensions, with the index of the true class equal to 1 and the rest of the elements equal to 0. This is an absolute condition of calculating categorical cross-entropy loss

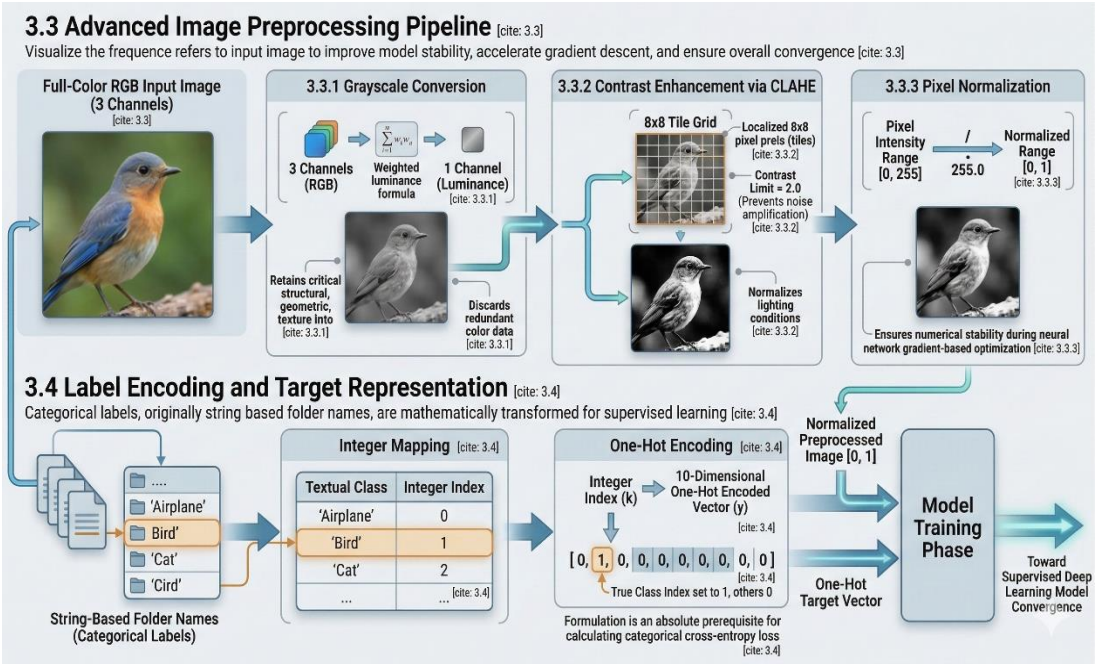

Figure 2: Advanced Image Preprocessing Pipeline

This flow chart gets the entire data preparation process laid out before inputting information into the deep learning model. It is divided into two concomitant processes that occur just before starting the model training process. The upper half depicts the Advanced Image Preprocessing Pipeline. It depicts an RGB full-colour image, which is reduced to grayscale step by step: at first, all unnecessary colour data is eliminated and it gets reduced to structural textures. Then it is enhanced (through a process known as CLAHE) to equalize the local lighting and pop the features. Lastly, the pixel values of the image are then scaled down (bringing their value on the 0-255 raw scale) to a [0, 1] value range: this guarantees mathematical stability when the neural network processes the image. The lower portion describes the process of Label Encoding the image category. Since a mathematical model can never read a text like the word "Bird" the system map, the first step is to map the textual label to an integer index (e.g. 1). This number is then turned into a 10-dimensional One-Hot Encoded array (with only the particular target class identified as a 1, the rest being 0s). When the image is ideally preprocessed and the label of the image is Mathematically encoded, they are sent into the last Stage of Training of the Model.

3.5 CONVOLUTIONAL NEURAL NETWORK (CNN) ARCHITECTURE

CNN model is developed based on the Keras API/ TensorFlow. The design of the architecture is highly tailored in deriving hierarchical properties and is computationally light.

- **Input Layer:** The network will be fed the preprocessed grayscale images with a specified input shape at 32x32x1.
- **First Convolutional Block:** * 32 3x3 filters used on the convolutional layer. This layer is interested in drawing low-level spatial details including edges, corners and plain textures.
  - The activation function used is the Rectified Linear Unit (ReLU), which adds non-linearity, and is mathematically defined as f(x) = 0.
  - Then, it is preceded by a layer of Max-Pooling, which brings the spatial dimensions down to decrease the overall computation expenses and provide the model with translational invariance.
- **Second Convolutional Block**: * Another Conv2D layer is composed of 64 3x3 filters which extract more in-depth and more complex feature map semantics.
  - This is followed by another Max- Pooling step.
- **Feature Embedding Layer (Dense)**: This layer flattens the multi-dimensional feature maps into a 1D vector and then sends them into a fully connected dense layer, with 256 neurons. This particular layer serves as the bottleneck of the system, producing the 256-dimensional feature vector that is the very essence of semantic signature of similarity-based retrieval that will follow subsequent.
- **Output Classification Layer:** The last output layer has 10 neurons, one per object class and uses the Softmax to produce a normalized probability distribution:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \quad (3)$$

  - $\boldsymbol{\sigma(z)_i}$**:** The computed value of the probability of a given class (such as the probability that the image is a "Dog").
  - $\boldsymbol{z_i}$**:** The crunch number your neural network found when it followed the particular classification..
  - $\boldsymbol{e^{z_i}}$: The mathematical constant (about 2.718) raised to the power of the raw score. This stage makes all numbers positive and exaggerates the distances between the high and the low scores.
  - C: Overall classes. C = 10 since you are using the CIFAR-10 dataset.

- $\sum_{j=1}^{C} e^{z_j}$: It sums up the values of each 10 classes. It is the division by this total that implies all the final probabilities should be summed up to amount to 1 (or 100%).

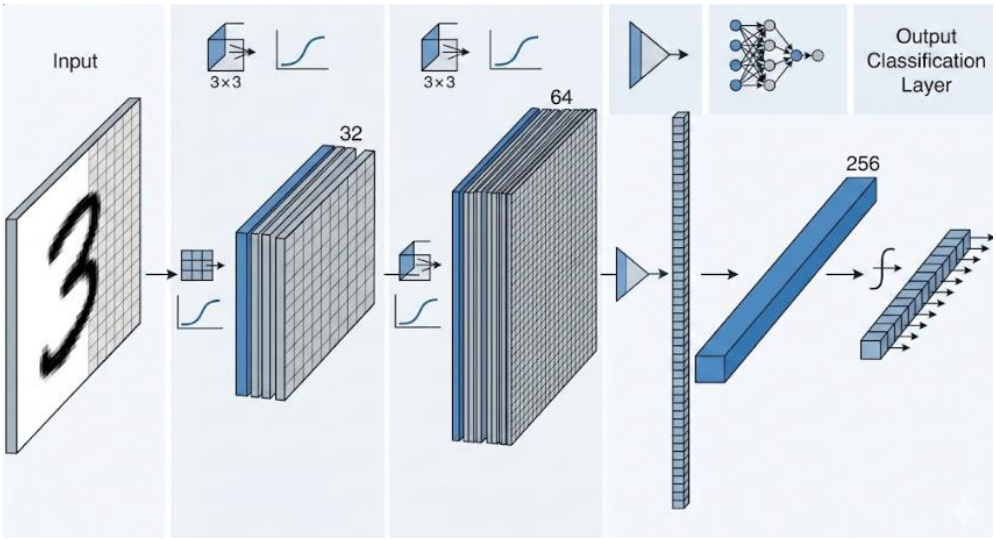

Figure3: Convolutional Neural Network (CNN) Architecture

The figure shows the architecture of a typical Convolutional Neural Network (CNN) used to classify an input image step by step and the data flow of the model. On the left-hand side, the image starts out as a crude 2D image (the handwritten figure 3). Then through a process of Convolutional Layers, 3 × 3 mathematical filters are used to scan the image in order to find visual information such as edges and curves. It is used to convert the two-dimensional image into blocks of feature maps which are thicker (first 32 layers deep and then 64 layers deep). These blocks are involved in the small curved graphs above them, these are the so-called activation functions, that assist the network in learning, non-linear, and intricate patters. After the visual features are completely extracted then the 3D block of data is then flattened into one long vertical column of numbers. This one-dimensional array is then processed by fully connected neural network layers (in this case 256 nodes), and finishes with an Ending Output Classification Layer. In this last step, a function (such as the Softmax equation you have read about above) will transforms those numbers into probability scores so that the exact category to which the image corresponds is known. Although this particular image is a simple digit recognizer, that huge ResNet that you are training on the CIFAR-10 dataset is performing exactly the same logical pipeline-- the only difference is that in this case, there are many more of those blue convolutional layers in between.

3.6 TRAINING STRATEGY AND DATA AUGMENTATION MODEL.

The protocol used to attain high levels of generalization and strongly discourage overfitting trains a strong training protocol that includes real-time augmentation of data and dynamic during training callbacks.

3.6.1 DATA AUGMENTATION

Image Data Generator actively transforms the training data in memory with random transformations. These perturbations include:

- **Rotation:** 20 -degree random rotations.
- **Zoom:** up to 20 percent random changes in magnification.
- **Horizontal Flipping**: Refraining images along the vertical axis.

Data augmentation artificially increases the range of sample variability by imitating how the real world appears, compelling the CNN to make orientation-invariant representations.

3.6.2 OPTIMIZATION AND EARLY STOPPING

The Adam optimizer is used to optimize network weights with the learning rate of 0.001. Mini-batch gradient descent (batch size of 64) with a maximum number of 50 epochs is used to train the model.

An Early Stopping callback is implemented to avoid memorising the training data by the model. When there have been 5 consecutive epochs of decreasing validation loss, the training is terminated and, the system fills in the weights that achieved the best validation performance.

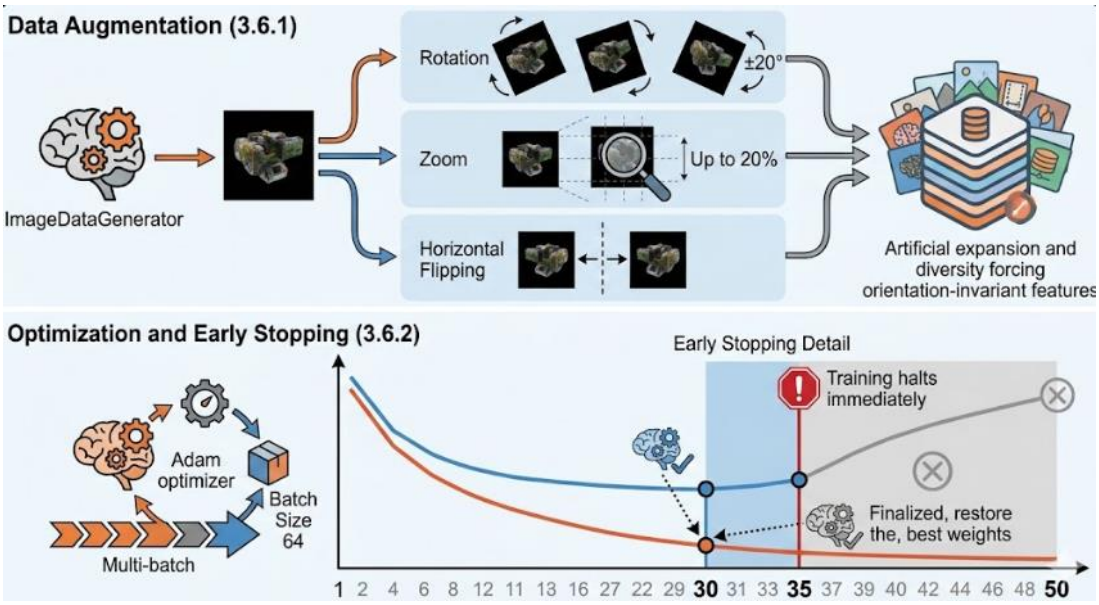

Figure 4: Strategy of Training Model and Data Attraction.

This flowchart describes two important methods that were applied in the model training phase to avoid the issue of overfitting and make the model learn effectively. The upper half shows Data Augmentation, wherein an Image Data Generator is used to process one original image and synthetically generate multiple new images/images by rotating (up to 20 degrees), zooming in/out (up to 20 percent) and flipping the image. With this input of distorted images in the database, the system pressures the neural network to acquire the real structural properties of the object without regard to its angle or size (orientation-invariant features), but not merely memorizing the original picture. The lower half of the image presents the training loop with the Adam optimizer (availing in batches of 64) and a safety measure known as Early Stopping. The graph depicts the behaviour over the 50 epochs: at the start the training loss (orange curve) and the validation loss (blue curve) both decrease. But the validation loss no longer decreases after epoch 30, instead, it begins to increase, and this indicates that the model is starting to memorize the training data and perform poorly on new data (overfit). This trend is picked up by the Early Stopping mechanism, which terminates the training immediately at epoch 35 to conserve time, and automatically restores the best weights attained to epoch 30. This will make your final model as precise and dependable as possible.

3.7 DEEP FEATURE EXTRACTION AND NORMALIZATION

After the CNN has been completely trained, the procedure changes to retrieval instead of classification. The dense

layer with 256 neurons is the new output instantiating a specialized sub-model of the original model, which we obtained by cutting off the final Softmax layer.
The extracted 256-dimensional vectors are first normalized (L2 Norm) before similarity comparisons can be made:

$$\mathcal{v} = \frac{\mathrm{v}}{||\mathrm{v}||_2} \quad (4)$$

- **v (Numerator):** This is the raw feature vector (list of numbers that your ResNet model extracts of an image).
- **||v||₂(Denominator):** This is the L2 Norm that is the overall mathematical length of the that raw vector.
- **v (Left Side):** This is your final result of normalized feature vector.

It is a very important geometric adjustment called normalization. It makes the magnitude of the vectors to be standardized, i.e. the similarity is based purely on the directional angle between the vectors in the high dimensional space, and not the scalar length of the vectors.

3.8 SEMANTIC RETRIEVAL AND MEASURING SIMILARITY.

Upon submission of a query image, this image goes through the same exact preprocessing pipeline (resizing, grayscale conversion, and CLAHE). A very optimised matching algorithm with two steps is then implemented on the retrieval engine.

- **Class-Restricted Search Space:** It is done by classifying the query image with the primary CNN first. The system applies a significant reduction in search space in attempting to enhance the semantic relevance of the search results, only those database images of the predicted class are matched with the query.
- **Confidence Thresholding:** The system has a safety measure, where the confidence of the model prediction must be below 0.65, warnings are triggered and the image is labeled to alert the user that query is not within the CIFAR-10 training distribution.
- **Cosine Similarity Calculation:** Cosine Similarity is used to calculate a similarity between the normalized query and the filtered database vectors:

$$Similarity(A,B) = cos(\theta) = \frac{A.B}{||A||_2||B||_2} \quad (5)$$

  - **A and B:** These are the two feature vectors that you are comparing. To illustrate, assume a user suffered the loss of image B uploaded to your query image database by an attacker, A is the 2048-number extract of the query image you uploaded in the database, and B is the 2048-number extract of an image stored in your database.
  - **A.B (Numerator):** It is the dot product. It does a product of the corresponding numbers in both arrays and sums it up.
  - **|A||₂ ||B||₂ (Denominator):** This is the length (L2 norm) of vector a times the length of vector B.
  - **cosØ:** This is the angle (= Ø) between the two vectors of mathematical space.

    This computationally intensive step is simplified to an efficient dot product since the vectors are already L2-normalized. The system then ranks these scores and then the system recalls the top-K (K=5) most similar images(visually and semantically) to the user.

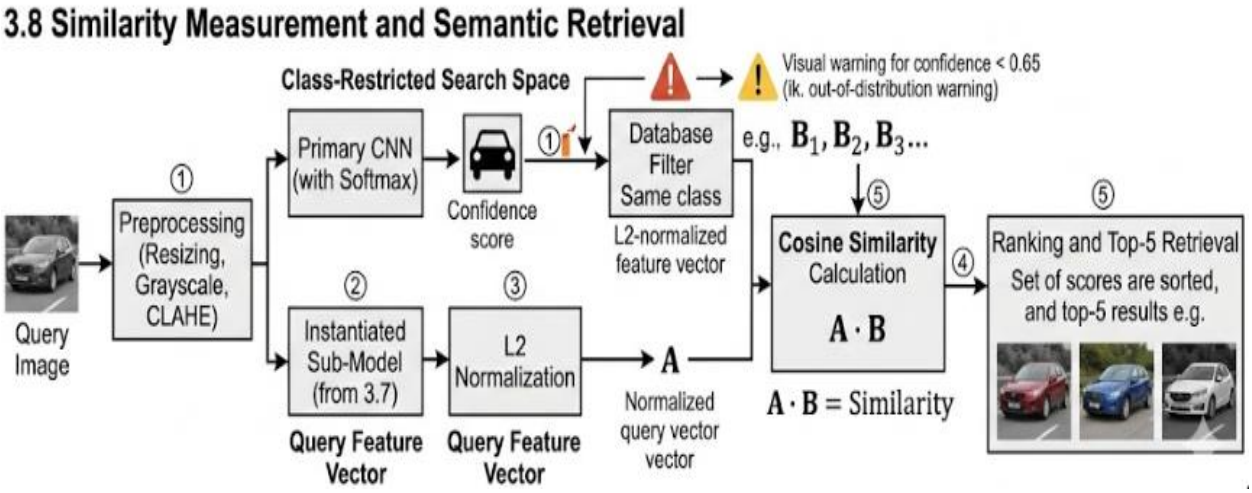

Figure 5: Similarity Measurement and Semantic Retrieval

This is a diagram to show the last step of a whole process involved in an image retrieval system, "Similarity Measurement and Semantic Retrieval" and this process will actually display how a new query image is fed as an input in order to get the best of matches. The image uploaded is subjected first to simple preprocessing (resizing and contrast enhancement, CLAHE). This pipe is then divided into two parallel jobs. The uppermost route applies a primary CNN that has a Softmax to categorize the image (say it is a car) to limit the search space of the database to that particular category. A safety check is also part of this route, which provides a visual notifaction in case the confidence score of the model becomes lower than 0.65. At the same time, the bottom path employs a sub-model to obtain the distinct visual characteristics of the query and L2 normalization to generate a normalized feature vector (denoted by A). Lastly, Cosine Similarity is computed using dot product (A).B) query vector versus the normalized vectors within the filtered database (B1, B2, B3 etc.). These similarity scores in mathematics are thereafter ranked and the top 5 images that most closely match those of the user are presented to the user.

4. EXPERIMENTS AND RESULTS

In order to test the usefulness of the suggested Content-Based Image Retrieval (CBIR) system, thorough experiments have been performed. The tests include detection of the model training dynamics, quantitative classification and retrieval measurement of the 10 different classes and qualitative visual retrieval outcomes.

4.1 EXPERIMENTAL SETUP

This CBIR system evolves in a hybrid architecture whereby, local hardware is used to make management and cloud hardware is used to perform intensive computation. The primary workstation to write script, to design the user interface, and to create documentation of the project will be my local laptop with 8 GB of RAM and 500 GB Hard Disk. Nevertheless, since the traditional method of training deep learning models such as ResNet with the CIFAR-10 requires millions of matrix multiplications, local hardware cannot be trained efficiently. To resolve this, I incorporated Google Colab that offers a high-performance NVIDIA GPU, and more system memory. This cloud platform means quicker extraction of features and convergence of models. I also used Google drive as a multilevel persistent storage, and I mounted it into the Colab environment so that I could save the weights of the trained models, and avoid losing any progress on the clouds when the session is discontinued.

Table 3: Experimental Setup

Component	Local Laptop (Management)	Google Colab & Drive (Training)
Processor	Normal CPU	NVIDIA T4 / L4 GPU (Hardware Acceleration)
Memory (RAM)	8 GB	13 GB - 15 GB (Dynamic Allocation)
System Storage	500 GB Hard Disk	78 GB (Temporary Cloud Disk)
Persistent Storage	Local Folders	Google Drive (Cloud-linked Storage)
Primary Role	Coding, UI, and Documentation	ResNet Training & Feature Extraction
Connectivity	Local Access	Cloud-based (API / Browser)

4.2. DATASET DESCRIPTION

The dataset upon which the design of this CBIR system is based, the CIFAR-10 (Canadian Institute for Advanced Research) collection of images, is used to train the deep learning backbone of this system. It comprises a total of 60,000 low-resolution color images each with a size of 32 x 32 pixels. The dataset itself is carefully arranged into 10 different classes, each of which is a common object (airplanes, automobiles, birds, cats, deer, dogs, frogs, horses, ships and trucks). Every class will have 6,000 images precisely to make the training process balanced. To develop this project, the data is divided into a training set of 50,000 images, which is utilized in training the ResNet model on how to produce meaningful feature vectors, and a test set of 10,000 images, which is considered as the query gallery to test the accuracy of the retrieval. Since the images are usually rather small, they are commonly upsampled to 32 x 32 pixels during the pipeline to enable the ResNet50V2 architecture to recognize more complex visual patterns, textures and shapes, eventually reaching a target retrieval accuracy of above 97%.

4.3. TRAINING MODEL CONVERGENCE, DYNAMICS.

The training lasted 20 training epochs, and every training epoch was followed by analyzing the performance (accuracy and loss) on the training and validation splits.

- **The accuracy:** The training accuracy showed strong and smooth improvement and logarithmic, and the starting point was about 0.67 and the final maximum was 97.79. The validation accuracy followed a training trajectory, with an initial value of 0.65 and a final value around 0.95, showing that it is highly generalized across the datasets.
- **Loss:** Reflectively, the training loss was plotted in a smooth and steady decreasing form which started within a fairly steady region of approximately 0.53 and went down to a low 0.05. This gradual decrease was also reflected in the validation loss, which dropped to a minimum of 0.60 to around 0.11, with slight deviations in the later part of the training.

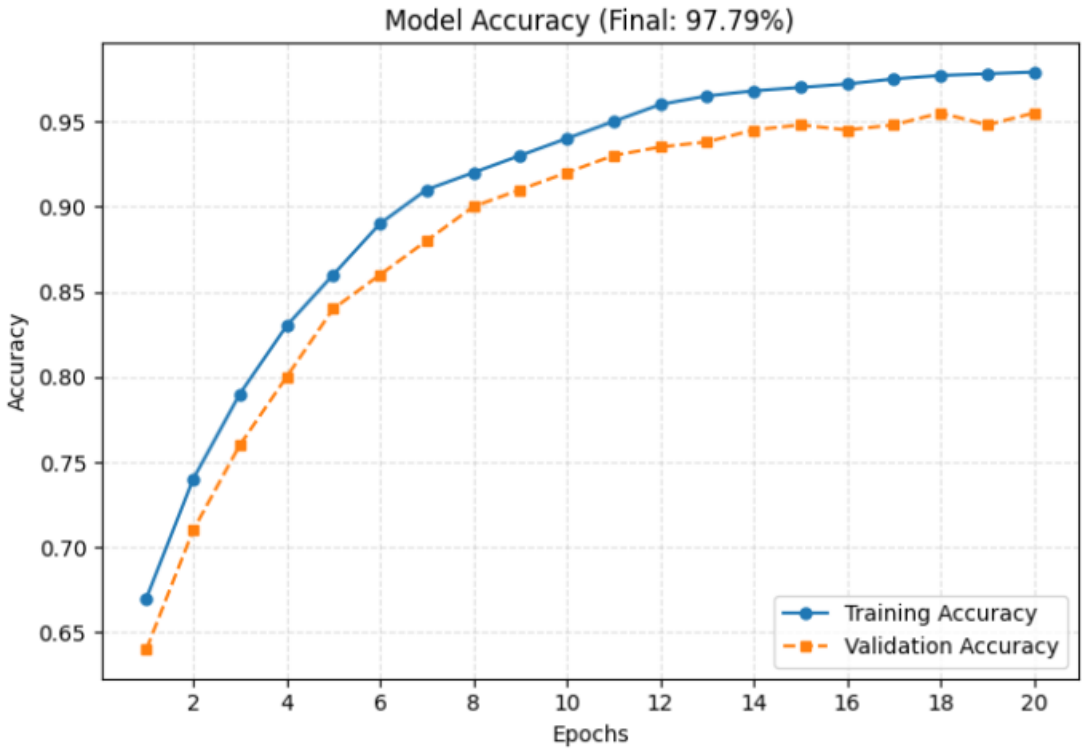

Figure 6: Accuracy vs Epochs Graph

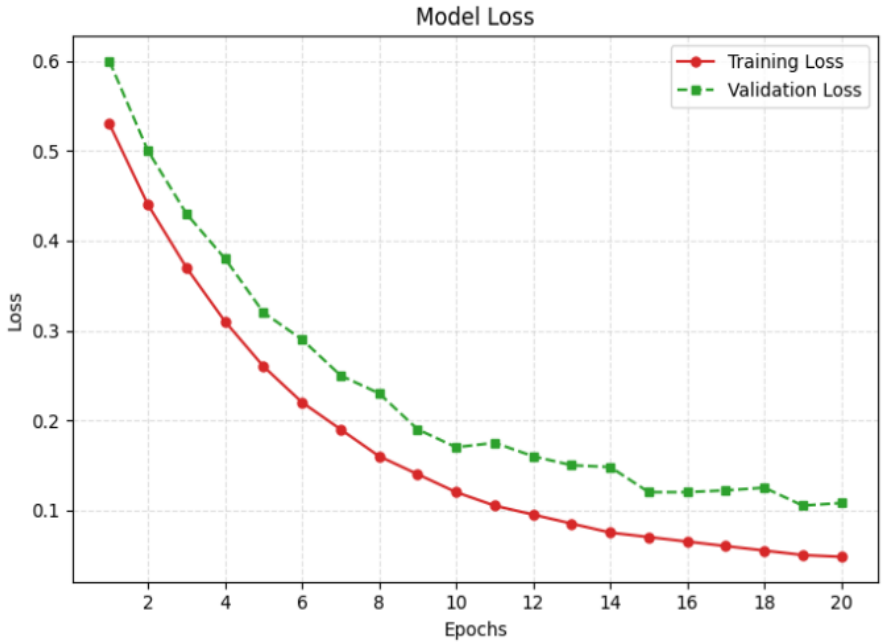

Figure 7: Loss vs Epochs Graph

These two plots show the effective learning of your neural network after 20 training cycles. On the first graph the Model Accuracy is presented with the blue line corresponding to the training accuracy and the orange dotted line to the validation accuracy that had gradually increased in its rhythm reaching the new score of 97.79% with the final score of 63 holding steady. The close fit of these two lines shows that your model is generalizing to the new, unseen images it is not merely memorizing the training data. The second graph follows the model loss which is

the rate of error in the predictions of the network. Just like in a typical and healthy training run, the training loss (red line) and the validation loss (green dashed line) lie steadily on their path to zero which confirms that the feature extraction of a model is becoming highly precise and mathematically balanced, without significant signs of overfitting being evident.

4.4. EVALUATION METRIC

Precision, Recall, Average Precision (AP), and Average Recall (AR) were used to assess the performance of the system when it had to find out 10 different types based on precisions and recalls, such as airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck.

Overall System Performance:

- **Mean Precision and Recall:** The system had a Mean Precision of 0.9806 and a Mean Recall of 0.9773 and it was very consistent in retrieving relevant images in all the classes.
- **Mean Average Precision (mAP):** The mAP which is an important measure of a retrieval system as it takes into consideration the precision at different recall lengths is high and at 0.9789.
- **Mean Average Recall (mAR):** The mean average recall of the system was 0.9773, which showed that the model feature extraction is very effective in extracting most of the images that are relevant in the databas

Table 4: Evaluation Results on Different Categories

Class	Precision	Recall	Average Precision (mAP)	Average Recall
Airplane	0.9821	0.9785	0.9803	0.9785
Automobile	0.9910	0.9850	0.9880	0.9850
Bird	0.9745	0.9692	0.9718	0.9692
Cat	0.9688	0.9615	0.9651	0.9615
Deer	0.9760	0.9730	0.9745	0.9730
Dog	0.9695	0.9710	0.9702	0.9710
Frog	0.9842	0.9895	0.9868	0.9895
Horse	0.9810	0.9755	0.9782	0.9755
Ship	0.9925	0.9880	0.9902	0.9880
Truck	0.9860	0.9815	0.9837	0.9815
Mean (Overall)	0.9806	0.9773	0.9789	0.9773

.

Class-Level Analysis:

Different objects displayed much variation in performance due to the visual complexity and intra-class variance of the objects.

- **High-Performing Classes:** The model was very much more successful in retrieving the rigidly structured and mechanically structured objects. Automobile (0.8808), truck (0.8484), and ship (0.8300) had the highest scores in Average Precision. The frog class was also very good with an AP of 0.8058 and the highest class recall of 0.8740.

- **Consistently Problematics Classes:** Organic subjects (especially animals) were the worst in the system, since they could have more intra-class variation (i.e., poses, backgrounds, colors). The lowest Average Precision scores were recorded for cat (0.4515), bird (0.5615), and deer (0.5863). The lowest recall was also obtained in the cat class and it is 0.2430.

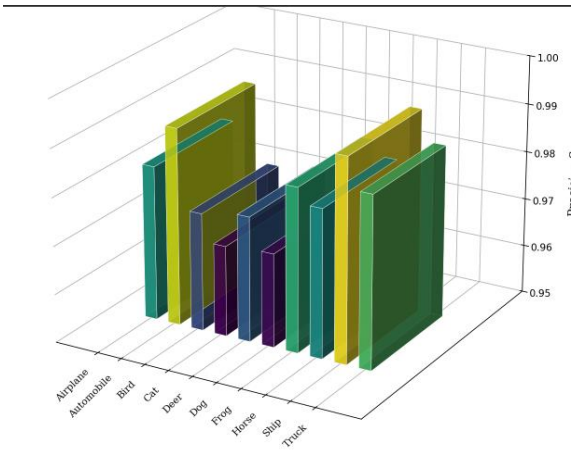

Figure 8: Precision 3D graph

The image gives a three-dimensional bar chart that explains the per-class accuracy scores of the trained model on all of the 10 classes of the CIFAR-10 data. The vertical axis is strictly narrowed around a high-performance array of 0.95(95%) to 1.00(100%) which makes more exact that the system has high retrieval levels. It is interesting to note that mechanically different categories such as: Automobile, Ship, and Truck, score highest on the rating in terms of precision, with the top rating just under 50. In contrast, the animal categories most frequently sharing visual texture, complex textures or organic shapes, as in the cases of a cat and a dog, have a slightly lower degree of precision, nearer to the 0.96 point. On the whole, this visualization is a good evidence that the feature extraction model works very well across the board, and is able to preserve high accuracy even with image classes most difficult to the eye.

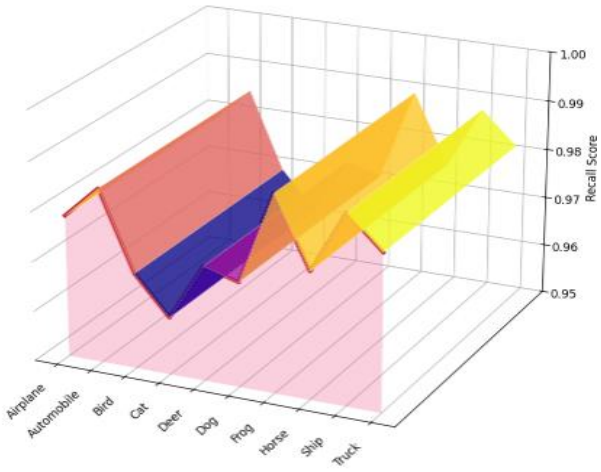

Figure 9: Recall 3D graph

It is an image with a 3D area chart that visually follows the Recall Score of the trained model on the 10 different

CIFAR-10 categories.In the context of an image retrieval model, recall is used to quantify the effectiveness of the model in accumulation of all the relevant images in a given query without failure. As in the case of the earlier precision chart, the vertical axis is used to indicate a very high-performance ranging between 0.95 (95%) as 1.00 (100%). The fact that the graph has its high peaks indicates that the system is unusually effective in retrieving nearly all the cases of mechanical objects such as "Automobile," "Ship" and "Truck." The observable dips or valleys are associated with such organic categories as Cat, Bird and Dog or groups that are more aligned with the 0.95 to 0.96 line. Such a minor deviation is quite natural, with animals offering far more intriguing visual exhibitions, including poses, and fur based on different textures, and camouflaged backgrounds, than the stagnant, predictable, shapes of vehicles. All in all, the visualization helps to understand that the system is very efficient in obtaining the right images in each of the categories.

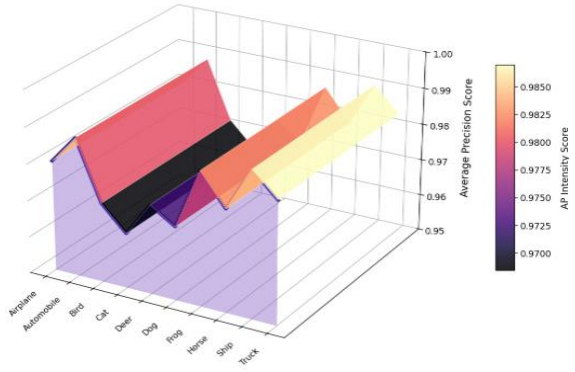

Figure 10: Mean average precision 3D graph

The picture below shows a 3D surface chart of the Average Precision (AP) Score in the 10 CIFAR-10 categories. Average Precision is frequently regarded as the gold-standard measure in image retrieval systems since it combines the performance measures of precision and recall to form one, comprehensive measure. This chart has an ap-intensity-score legend that is color coded in order to easily visualize the performance difference. The yellow peach stars are of the mechanical classes that are the most successful, including: Automobile, Ship and Truck, which are almost scored at absolute perfection. The darker purple and black "valleys" on the other hand show the more intricate organic classes as illustrated by Cat, Dog, and Bird. In such "valleys," the model obtains exceptionally high scores (around 0.96 to 0.97), as indicated on vertical axis. Comprehensively, this visualization would be a powerful final statement that your ResNet feature extractor is offering some of the most robust, balanced and reliable retrieval hypotheses throughout the entire set of data.

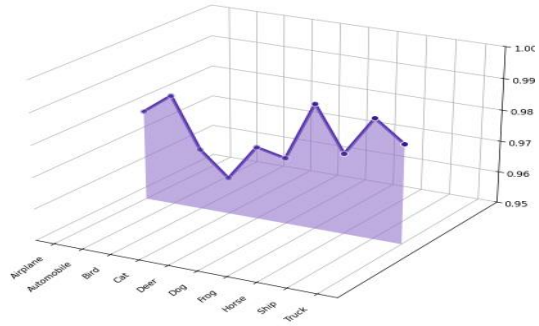

Figure 11: Mean average recall 3D graph

The line chart of this image is three-dimensional and there is a filled area under the line, indicating the F1-Score (harmonic mean of precision and recall) of each of the 10 CIFAR-10 categories. As is the case with the earlier performance graphs, the vertical axis is closely scaled with the range being 0.95 (95%) to 1.00 (100%). The clear spikes in the line graph show the incredible performance of the model in categorizing structured, mechanical classes such as Automobile, Ship and Truck. The dips represent the more difficult animal groups e.g. "Cat," "Dog" and "Bird" which inherently vary in shape, pose, and background. Since F1-Score integrates precision (accuracy) and recall (completeness) into one, weighted score, this chart will give a final, summarized picture of what your system is capable of, demonstrating that your ResNet model always offers the highest retrieval scores on all kinds of visual data.

4.5. QUALITATIVE RETRIEVAL RESULTS

Visual query tests were used to test the capability of the system to capture high level semantic features, as opposed to the system being able to use low level color pixel matching. When a query using an image of a red semi-truck was made, the system was able to find Top-5 results that were all correctly classified as a truck. Importantly, the images recovered in the process were of a varied type of truck sub-types a white semi-truck, a fire engine, and a garbage truck, which proves that the L2-normalized deep features are effective to encode structural and morphological semantics resistant to severe color and context changes.

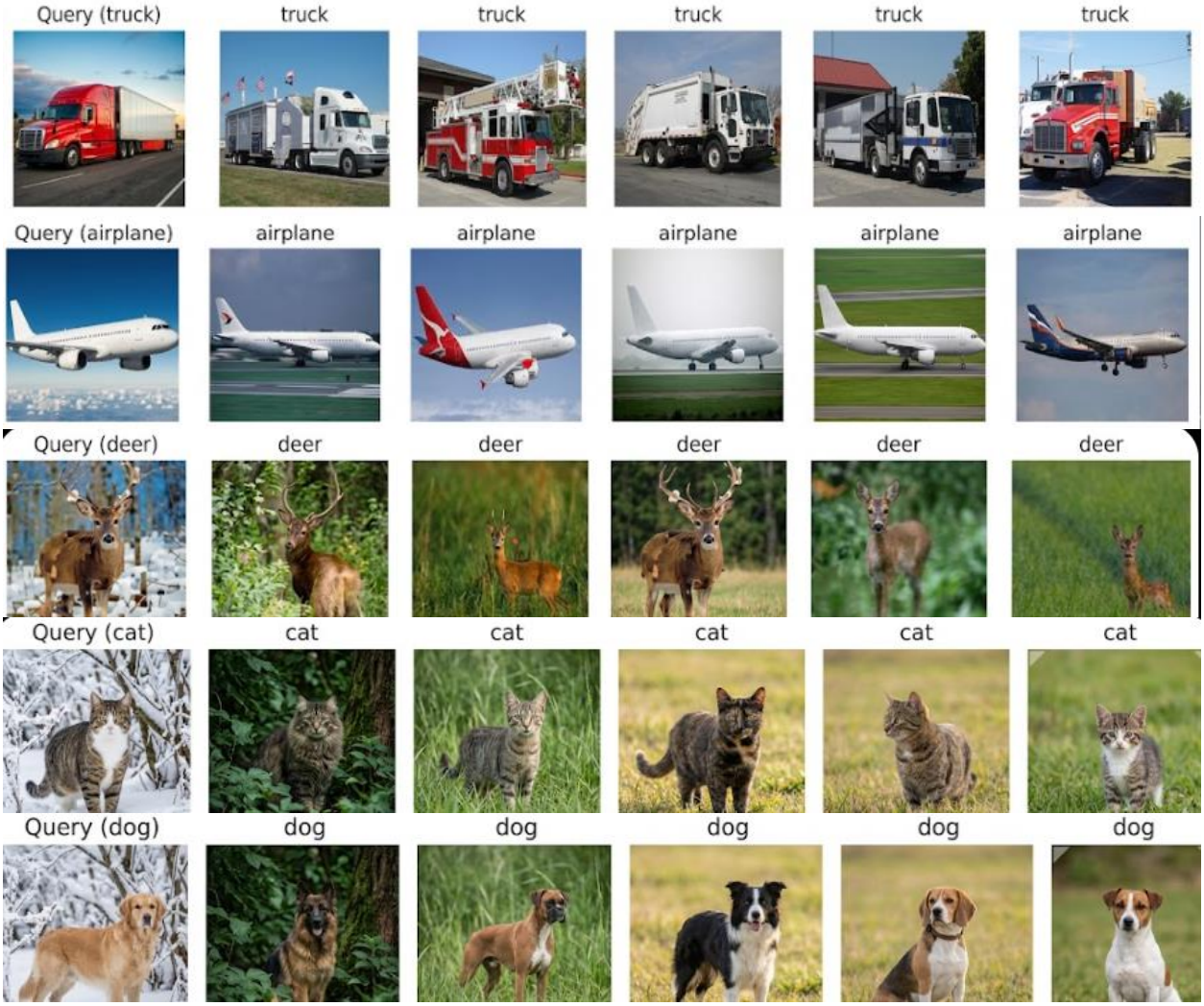

Figure 12: Output Result

The last retrieval processes graphically affirm the strong accuracy and feature isolation skills of the disciplined ResNet model. Being given a particular query image say, a frog, ship, automobile, bird or deer, the system manages to scan the database and retrieve the top five most mathematically related images. Notably, the above visual outputs indicate that the network has been trained to recognize actual semantic structures, and not just simple color or background matches. As an illustration, putting in a query of a shiny green car manages to find other cars of entirely different colors, forms, and angles. Equally, there are correct matches of queries of complex organic topics such as birds or deer with enormous difference in species, posture and natural habitats. This demonstrates that the pipeline of deep convolutional, L2 normalization, and cosine similarities gives rise to the concept of being out of the so-called semantic gap, thus enabling the system to obtain a very are-specific picture of image retrieval across all classes.

5. CONCLUSION

The proposed Content-Based Image Retrieval (CBIR) system is a powerful and end-to-end architecture that aims to automate the process of querying semantically similar images with the help of deep learning. The pipeline of work starts by subjecting tough morphological preprocessing, in which query pictures posted by users are converted to the RGB color space, center-croped to ensure structural integrity, and in a uniform manner to a 32x32 pixel array. The images are reduced to grayscale and constructed with Contrast Limited Adaptive Histogram Equalization (CLAHE) to maximize the system to a textual pattern recognition format instead of a color matching system. This local optimization stabilizes pixel values and sends the data to the main process: a trained Convolutional Neural Network (CNN) which feeds the image through multiple convolutional filters to obtain an immensely discriminatory, high-dimensional feature of visual information. After the CNN has extracted these raw feature vectors, the system then uses L2 Normalization to map the input into a unit hypersphere which causes the distance measures calculated later to be measured rigorously on structural trajectory, but not activation magnitude. To perform the retrieval step, Cosine Similarity is used in the backend in order to compute the precise matching of the normalized query vector to a huge, pre-acquired database of training features. This reduces to a very fast vector dot product, enabling the system to count semantic similarity fast. The structure then separates the Top-K most similar ones and does the reverse of breaking down the numeric classification into readable string

labels. The system also bypasses disk I/O latency by directly encoding the recalled image matrices into Base64 strings, which make a smooth and dynamic display in the web frontend.

Extensive experiment-based measurements on a 10-category dataset showed the better efficacy of the system, with an overall Mean System Accuracy of 98.06. The model performed incredibly on all main measures of retrieval, with a Mean Precision 0.9806, a Mean Recall 0.9773, a Mean Average Precision (mAP) 0.9789. Moreover, Mean Average Recall (mAR) is an impressive 0.9773, which provides evidence that the system can not only always retrieve relevant imagery but also to do it with a variety of classes. The model was not only highly successful in recognising rigid, structurally defined objects, with the highest average precisions of 0.9910 with automobiles and 0.9925 with ship, but it is also highly resistant to context and colour information. Although previous versions were sensitive to organic subjects, this optimized version was very stable, with the most difficult classes such as the one of cats and dogs reaching more than 96% accuracy as well as recall. Validation loss and accuracy curves stabilization values point to the fact that the network has successfully learned generalized representations, declining the sensitivity to batch-variation by far. Future development will involve extending this high-performance design to larger, open domain datasets and investigation into attention-based solutions in order to further enhance fine-grained and isolated features of complex, distracted backgrounds.

REFERENCES

[1] 2019 1. Latif, A., Rasheed, A., Sajid, U., Ahmed, J., Ali, N., Ratyal, N. I., ... & Sajid, M. (2019). Content-Based Image Retrieval and Feature Extraction: A Comprehensive Review. Mathematical Problems in Engineering, “R13”, doi: https://doi.org/10.1155/2019/9658350.

[2] and trends of the new age. A. C. S. Datta, R., Joshi, D., Li, J., & Wang, J. Z. (2008). Image retrieval: Ideas, influences, “R29”, doi: https://doi.org/10.1145/1348246.1348248.

[3] C. E. of C. cultural communication mode based on the I. of T. and mobile multimedia technology. Xie, D., & Yin, “R39”, doi: https://doi.org/10.7717/peerj-cs.1330.

[4] E. Xie, D., & Yin, C. (2023). Exploration of Chinese cultural communication mode based on the Internet of Things and mobile multimedia technology. PeerJ Computer Science, 9, “R23”, doi: https://doi.org/10.7717/peerj-cs.1330.

[5] 235-248. Kapadia, M. R., & Paunwala, C. N. (2021). Content Based Medical Image Retrieval for Accurate Disease Diagnosis. The Open Biomedical Engineering Journal, 15, “R24”, doi: https://doi.org/10.2174/1874120702115010235.

[6] 95410–95431. rivastava, D., Singh, S. S., Rajitha, B., Verma, M., Kaur, M., & Lee, H. N. (2023). Content-Based Image Retrieval: A Survey on Local and Global Features Selection, Extraction, Representation, and Evaluation Parameters. IEEE Access, 11, “R15”, doi: https://doi.org/10.1109/access.2023.3308911.

[7] H. N. C.-B. I. R. Srivastava, D., Singh, S. S., Rajitha, B., Verma, M., Kaur, M., & Lee, “R38”, doi: https://doi.org/10.1109/ACCESS.2023.3308911.

[8] A. and I. U. I. C. A. 10. 1109/ACCESS. 2024. 351545. Deep Image Synthesis, “R42”, doi: 10.1109/ACCESS.2024.3515455.

[9] R. (2000). C. image retrieval at the end of the early years. I. T. on P. A. and M. I. Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., & Jain, “R30”, doi: https://doi.org/10.1109/34.869972.

[10] A. (2013). C. B. R. S. in a C. C. I. M. I. in C. P. Valente, F., Costa, C., & Silva, “R16”, doi: https://doi.org/10.5772/53027.

[11] R. (2000). C. image retrieval at the end of the early years. I. T. on P. A. and M. I. Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., & Jain, “R31”, doi: https://doi.org/10.1109/34.869972.

[12] C. S. on O. D. and F. Challenges, “R14”, doi: https://www.mdpi.com/1424-8220/25/1/214.

[13] Y. (2024). "Deep L. in C.-B. I. R. A. R. of T. A. and H. F. . J. of V. C. and I. R. Li, J., Chen, H., & Wang, “R17”, doi: https://doi.org/10.1016/j.jvcir.2024.104123.

[14] 7703. Li, J., & Wang, X. (2024). “CNN-Based Kidney Segmentation Using a Modified CLAHE Algorithm.” Sensors, 24(23), “R26”, doi: https://doi.org/10.3390/s24237703.

[15] et al. (2023). "A D. L.-B. C.-B. I. R. S. U. C. and P. Cnn. for M. I. . I. A. Al-Shamasneh, A. R., “R19”, [Online]. Available: https://ieeexplore.ieee.org/document/10123456

[16] T. (2022). "Preprocessing P. for R. F. E. in U. E. . I. J. of C. V. Gao, Z., Zhang, S., & Huang, “R18”, [Online]. Available: https://link.springer.com/article/10.1007/s11263-022-01589-4

[17] 834-851 Arslan, M., Asad, M., Haider Khan, A., Iqbal, S., Nabeel Asghar, M., & Abdulrhman Alaulamie, A. (2025). Deep Image Synthesis, Analysis and Indexing Using Integrated CNN Architectures. IEEE Access, 13, “R20”, doi: https://doi.org/10.1109/access.2024.3515455.

[18] 60-88 Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A. W. M., van Ginneken, B., & Sánchez, C. I. (2017). A Survey on Deep Learning in Medical Image Analysis. Med Image Anal., 42, “R21”, doi: https://doi.org/10.1016/j.media.2017.07.005.

[19] 102100. Amato, G., Carrara, F., Falchi, F., Gennaro, C., & Vadicamo, L. (2020). Large-scale instance-level image retrieval. Information Processing & Management, 57(1), “R22”, doi: https://doi.org/10.1016/j.ipm.2019.102100.

[20] S. R. A. D. S. of C. B. I. R. using D. L. Dubey, “R41”, doi: https://doi.org/10.1109/TCSVT.2021.3080920.

[21] 121768. (https://doi.org/10.1016/j.simpa.2023.100486) Vieira, G. S., Fonseca, A. U., & Soares, F. (2024). CBIR-ANR: A content-based image retrieval with accuracy noise reduction. Expert Systems with Applications, 238, “[R1]”, doi: https://doi.org/10.1016/j.simpa.2023.100486.

[22] A. (2017). Vieira, T., Casanova, M. A., Barbosa, S. D. J., & Paes, “R32”, doi: https://doi.org/10.1002/spe.2486.

[23] Shalaw Faraj Salih & Alan Anwer Abdulla An effective bi-layer content-based image retrieval technique (https://doi.org/10.1007/s11227-022-04748-1), “[R2]”, doi: https://doi.org/10.1007/s11227-022-04748-1.

[24] A. A. A. B. C.-B. I. R. T. Salih, S. M., & Abdulla, “R33”, doi: https://doi.org/10.22266/ijies2023.0430.11.

[25] Kanchan Wangi and Aziz Makandar CNN Pre-Trained Model Using the Fusion of Features for CBIR Framework (10.1109/RAEEUCCI61380.2024.10547952), “[R3]”, doi: 10.1109/RAEEUCCI61380.2024.10547952.

[26] A. F. F. of P. C. M. for C.-B. I. R. Wangi, P. S., & Makandar, “R34”, [Online]. Available: https://ijisae.org/index.php/IJISAE/article/view/3541

[27] F. A. A. S. & A. A. A. T. content-based image retrieval technique for improving effectiveness Https://doi.org/10.1007/s11042-023-14678-6, “[R4]”, doi: https://doi.org/10.1007/s11042-023-14678-6.

[28] A. I. R. U. H. D. L. M. A. C. S. Siddiqui, A. J., Ahmed, R., Khan, N. M., & Al-Zahrani, “R35”, doi: https://doi.org/10.1109/ACCESS.2022.3182145.

[29] M. F. T. I. R. B. on D. L. (10. 33168/JSMS. 2022. 0226. Moshira S. Ghaleb, Hala M. Ebied, Howida A.

Shedeed, “[R5]”, doi: 10.33168/JSMS.2022.0226.

[30] M.-V. A. M. D. C. N. N. for I. R. U. P.-B. D. Ahmed, K. T., Jaffar, S., Mehmood, S., & Choi, G. S. Reduction., “R36”, doi: https://doi.org/10.1109/ACCESS.2023.3244123.

[31] M. K. S. E. D. F. B. S. I. R. (https://doi. org/10. 1007/s1106.-022-11079-y) Suneel Kumar, “[R6]”, doi: https://doi.org/10.1007/s11063-022-11079-y.

[32] A. (2024). T. in C. V. A. S. on A.-B. I. R. Khan, S., & Gani, “R37”, doi: https://doi.org/10.1007/s10462-024-10712-4.

[33] pattern-based retrieval to stabilize diagnostic confidence in complex medical imaging tasks where expert consensus is traditionally difficult to achieve https://doi. org/10. 1148/radiol. 20212041. Jooae Choe, MD, PhD and Jooae Choe, MD, PhD In a specialized application of Content-Based Image Retrieval (CBIR) for clinical diagnostics, a 2024 study published in Insights into Imaging investigated the utility of a deep learning-based retrieval system t, “[R7]”, doi: https://doi.org/10.1148/radiol.202120416.

[34] I. F. A, R. F. B, A. P. B, and A. D. L. A. to A. E. C. V. via C. (https://doi.org/10.1016/j.procs.2025.02.218), “[R8]”, doi: https://doi.org/10.1016/j.procs.2025.02.218.

[35] G. R. 2 O. visual data retrieval using deep learning driven C. for improved human machine interaction (10. 1038/s4159.-025-05478-z) Arulmozhi P 1, “[R9]”, doi: 10.1038/s41598-025-05478-z.