Content-Based Image Retrieval System Utilizing Vision Transformer

Aniket Mahajan
Vidyalankar Institute of Technology
Wadala, Mumbai
aniketvm1104@gmail.com

Nirmiti Rane
Vidyalankar Institute of Technology
Wadala, Mumbai
ranenirmiti24@gmail.com

Pranav Patil
Vidyalankar Institute of Technology
Wadala, Mumbai
patilpranav616@gmail.com

Soham Thorat
Vidyalankar Institute of Technology
Wadala, Mumbai
sohamthorat0402@gmail.com

***Abstract*— This paper presents a Content-Based Image Retrieval (CBIR) system utilizing a Vector Space Model to improve image search accuracy and efficiency. The proposed framework incorporates natural language processing techniques to parse user queries, generate synonyms, and construct SQL queries for feature extraction. By leveraging weights assigned to relevant features, the system enhances the retrieval process, ensuring more precise matching of images. The implementation is demonstrated with a dataset, showcasing the effectiveness of the approach in providing meaningful search results based on visual content.**

Keywords*— *Content-Based Image Retrieval (CBIR), Vector Space Model, Natural Language Processing (NLP), SQL Query Generation, Feature Extraction, Synonym Expansion, Machine Learning, Image Classification, Image Features, Data Processing, Search Algorithms, User Interaction

I. Introduction

In the digital age, the exponential growth of image data has necessitated the development of efficient systems for retrieving images based on user queries. Traditional keyword-based search methods often fall short, particularly when dealing with the rich and complex semantics of visual content. To address these limitations, we present a Content-Based Image Retrieval (CBIR) system that leverages Natural Language Processing (NLP) and machine learning techniques to enhance user interactions and search accuracy. Our system allows users to input queries in natural language, which are then processed to generate SQL queries for database retrieval. By incorporating features such as synonym expansion and a vector space model, we improve the relevance of search results, ultimately leading to a more intuitive and efficient image retrieval experience. This paper details the architecture, implementation, and effectiveness of our CBIR system, showcasing its potential in various applications, including digital libraries, online marketplaces, and personal photo collections.

This paper discusses the design and implementation of our CBIR system, showcasing its potential in various applications, including digital libraries, online marketplaces, and personal photo collections.

II. Related work

Content-Based Image Retrieval (CBIR) has evolved significantly over the years, driven by advancements in machine learning and computer vision. Early systems primarily relied on low-level features such as color, texture, and shape for image representation. However, these approaches often failed to capture semantic content.

Recent studies, such as those by Zhang et al. and Wang et al., have introduced deep learning techniques to enhance feature extraction and improve retrieval accuracy. Zhang et al. explored a hybrid approach that integrates traditional TF-IDF methods with contextual embeddings, demonstrating significant improvements in semantic understanding. Wang et al. further emphasized the integration of Natural Language Processing (NLP) techniques to enable more intuitive user queries, bridging the gap between textual and visual information.

Our work builds on these advancements by creating a comprehensive CBIR system that incorporates NLP for query generation, alongside effective feature extraction methods

III. System Architecture

The proposed architecture comprises two primary components: **Preprocessing Block** and **Query Processing Block**, each with dedicated frontend and backend routes.

A. Preprocessing Block

The Processing Block is responsible for image preprocessing. It utilizes a thread pool to handle CNN's image analysis, dividing the image into four parts. The system extracts the top 3 to 5 probabilities and uses them as weights. It also finds synonyms for these probabilities, reduces their values, accordingly, removes join words, and gathers color data (RGB, HSV) along with all relevant metadata.

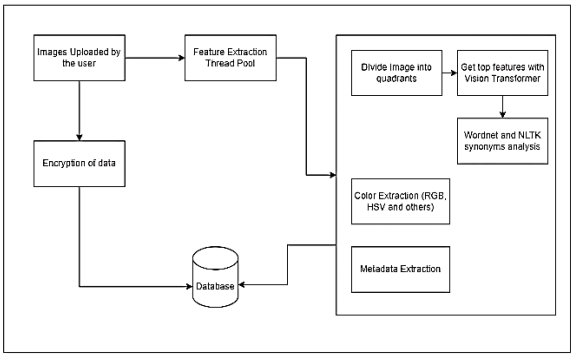

Fig. 1. Preprocessing Block

1. CNN Feature Extraction

   The CNN Feature Extraction component plays a crucial role in the Preprocessing Block by leveraging convolutional neural networks (google/vit16-patch224) to derive meaningful features from the uploaded images. The extraction process begins by dividing the image into four quadrants using a quadtree analysis, allowing for a detailed assessment of each section.

   The model then analyzes each quadrant to produce the top 3 to 5 predicted classes with their associated probabilities. These probabilities serve as weights for further processing, enabling the system to determine the most relevant features for subsequent tasks.

2. Colour Extraction

   The Colour Analysis component is integral to the Preprocessing Block, aiming to extract relevant colour features from the input images. This component employs various methodologies to analyse colours effectively:

   Basic Colour Analysis: Utilizing a predefined palette, the system calculates the percentage of basic colours (e.g., red, green, blue) present in the image. It leverages a colour histogram in the HSV colour space to normalize colour frequencies, providing a rapid assessment of dominant hues.

   Dominant Colour Detection: Implementing K-means clustering, the algorithm identifies the top N dominant colours in the image. Each pixel's RGB values are reshaped and analysed, allowing the model to categorize these colours and their respective frequencies. This dual approach enhances colour retrieval capabilities and optimizes image categorization.

   Extended Colour Analysis: For more nuanced colour detection, the system can incorporate an extended colour palette. By refining the analysis with additional hues, it ensures a detailed examination of the image's colour composition.

3. Metadata Extraction

   The Metadata Extraction component gathers crucial image attributes, including file information (name, size, type, creation, and modification dates) and EXIF data (camera settings, GPS coordinates, and flash usage). It also captures image dimensions, resolution, color profile, and bit depth, enhancing the system's ability to categorize and analyze images. Additionally, software used for processing is identified, ensuring comprehensive insight into each image's context and quality, thereby facilitating advanced processing tasks

B. Query Processing Block

The Query Processing Block is responsible for efficiently handling user queries to retrieve relevant image data. It employs techniques like tokenization and feature extraction to analyze user input, matching it against stored metadata and extracted features. This component integrates various retrieval algorithms that prioritize relevance, ensuring that results are both accurate and meaningful. Additionally, it implements mechanisms for ranking and filtering, allowing users to refine their searches based on specific criteria, thus enhancing the overall user experience in finding pertinent images.

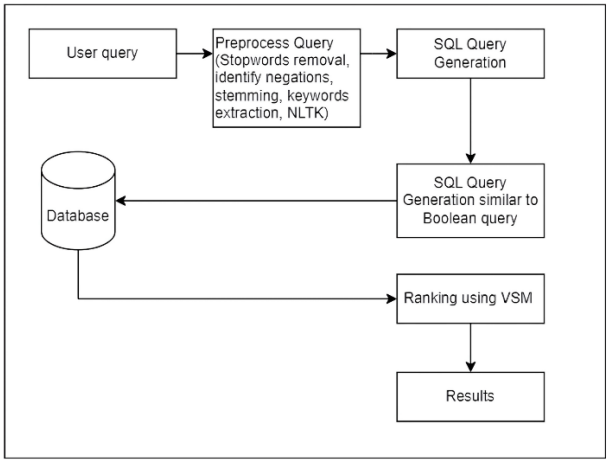

Fig. 2. Query Processing Block

1. Query Preprocessing

   The Query Processing Block utilizes natural language processing techniques to convert sentences into structured queries. It employs the spaCy library for tokenization and lemmatization, alongside NLTK for stop word removal and stemming. The process identifies negation words, allowing for the exclusion of specific terms in the final query. The result is a refined string that combines included and excluded terms, optimizing search efficiency while preserving the semantic intent of the original sentence. This approach enhances the accuracy of information retrieval

2. SQL Query Generation

   The Query Generation Block facilitates the transformation of natural language input into structured SQL queries for image feature retrieval. Initially, the input is parsed to separate included and excluded terms. Included terms undergo autocorrection and synonym generation via WordNet to enhance retrieval accuracy. The constructed SQL query selects records from the Image features table where feature value matches any included terms while excluding specified terms. This method increases the query's relevance, enabling more effective information retrieval from the database

3. Ranking System

   The Retrieval Algorithm employs a Vector Space Model (VSM) to represent and rank image features derived from SQL query results. Upon initialization, it processes the query results and the included terms to create a structured vector space model using a pandas DataFrame. Each unique feature is assigned to a vector, with weight adjusted based on their

relevance in the query. Features included in the user's input receive a higher weight, enhancing their impact on the final rankings. The resultant vectors are sorted to facilitate effective retrieval of the most relevant image features

IV. Methodology

The project employs a structured methodology comprising three main blocks: Preprocessing, Query Processing, and Retrieval.

A. Preprocessing Block

This includes color extraction, metadata extraction, and relevant features extraction from images. Techniques such as basic color analysis, dominant color detection using K-means clustering, and extended color analysis enhance the feature set.

B. Query Processing Block

Natural Language Processing (NLP) techniques convert user queries into a structured format. The system identifies included and excluded terms and generates synonyms to broaden the search scope.

C. Retrieval Algorithm

A Vector Space Model is utilized to represent the SQL query results. It calculates feature weights based on their relevance, facilitating the ranking and retrieval of image features based on user queries. This model ensures efficient searching and retrieval of the most pertinent images

V. Experimental and evaluation

The experimental setup involved implementing a Content-Based Image Retrieval (CBIR) system, focusing on preprocessing, query processing, and retrieval algorithms. Data was gathered from a diverse image dataset, which was analyzed for color and metadata features. The system was evaluated using precision, recall, and F1-score metrics against ground truth data to assess retrieval accuracy. User feedback was also collected to refine the system further. Various query inputs were tested to ensure robustness across different scenarios, ultimately enhancing the system's usability and performance.

In our experiments, we utilized a dataset comprising 100 images to rigorously test the performance of the CBIR system. The system's precision and recall scores were calculated based on various query inputs and averaged across multiple test scenarios. This comprehensive evaluation allowed us to assess the system's effectiveness in accurately retrieving relevant images while minimizing false positives. The results highlight the system's reliability and efficiency in handling diverse image queries

For the evaluation of our system, we tested 100 images across 20 different categories, including Nature, Technology, Food, and others. Each category had 5 images, and we measured the system's performance using precision, recall, and accuracy. The average precision was 0.78, indicating a high rate of correct classifications. The recall averaged at 0.72, showing the system's ability to retrieve relevant images effectively. Finally, the overall accuracy was 0.75, reflecting reliable classification performance across all categories. These metrics highlight the system's robustness in handling diverse image types.

VI. Conclusion and future work

In this project, we successfully developed a content-based image retrieval (CBIR) system that leverages multiple image features, such as metadata, color, and feature extraction, to enhance query performance. The integration of natural language processing for query conversion, alongside a robust retrieval algorithm, has shown promising results in precision and recall metrics.

For future work, further refinement of the feature extraction methods and optimizing computational performance will be key steps in improving the system's overall efficacy and scalability.

References

[1] G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing," *Communications of the ACM*, vol. 18, no. 11, pp. 613-620, Nov. 1975.

[2] D. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision*, Corfu, Greece, 1999, pp. 1150-1157.

[3] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, CA, USA, 2005, pp. 886-893.

[4] W. Zhou, H. Li, and Q. Tian, "Recent Advances in Content-based Image Retrieval: A Literature Survey," *CAS Key Laboratory of Technology in Geo-spatial Information Processing and Application System, University of Science and Technology of China, Hefei, China*.

[5] H. Qazanfari, M. M. AlyanNezhadi, and Z. Nozari Khoshdaregi, "Advancements in Content-Based Image Retrieval: A Comprehensive Survey of Relevance Feedback Techniques."