CLIP-DB: A Database-Backed Semantic Image Search Engine

Hairui Yin
University of Maryland
College Park, MD 20742, USA
yinhr@umd.edu

Abstract

Traditional image search engines typically operate in two distinct modes: image-to-image and text-to-image retrieval. In image-to-image search, systems rely on techniques such as SIFT, feature bundling, or hash functions to extract low-level visual features (e.g., edges, textures) and then perform similarity matching against a database. Meanwhile, text-to-image search often depends on keyword matching between query text and associated metadata (e.g., filenames, URLs, or captions), disregarding the actual visual content. A key limitation of these approaches is their inability to capture semantic features—the deeper meaning and context of images beyond superficial attributes. To address this gap, we propose a unified model leveraging Contrastive Language-Image Pre-Training (CLIP) to encode images or text into a related embedding space. By converting queries into embeddings, our system enables semantic similarity comparisons against a preprocessed database of image embeddings. This approach bridges the gap between visual and textual modalities and ensures retrieval results align with human content based understanding. Our method offers a robust solution for semantic-aware image search. The code implementation for this project is available on GitHub at https://github.com/henryparker/Clip_DB.

1 Introduction

Image search has long been a fundamental task in computer vision and information retrieval, enabling users to find relevant images from vast databases. Traditional image search engines typically employ two primary approaches: image-to-image retrieval and text-to-image retrieval.

Image-to-image retrieval systems commonly rely on extracting low-level visual features, such as edges, textures, and shapes, using techniques like SIFT, feature bundling, or hash functions. Similarity matching is then performed based on these extracted features to find visually similar images within a database. Conversely, traditional text-to-image search methods often depend on keyword matching between the user's text query and associated metadata of images, such as filenames, URLs, or captions. While straightforward, this approach frequently overlooks the actual visual content of the image, leading to potential mismatches when the metadata does not accurately reflect the image's subject or semantic meaning.

A significant limitation inherent in both of these traditional approaches is their struggle to effectively capture and understand the semantic content of images—the deeper meaning and context that goes beyond superficial visual attributes or simple keyword associations. This often results in retrieval results that may be visually similar or match keywords but fail to align with the user's true intent or the semantic relevance of the image.

To address this critical gap, this project proposes a unified approach for semantic-aware image search leveraging the power of Contrastive Language-Image Pre-Training (CLIP). Our method aims to develop a system capable of understanding the deep semantics of both images and text, enabling unified cross-modal retrieval for more accurate image search. The core idea is to map images and text into a shared embedding space where the distance between embeddings can effectively measure their semantic similarity. By encoding queries (either image or text) into this embedding space, our system can perform semantic similarity comparisons against a preprocessed database of image embeddings, ensuring retrieval results are more aligned with human content-based understanding. This approach offers a robust solution for enhancing the accuracy and relevance of image search by focusing on semantic understanding.

We summarize our main content as follows:

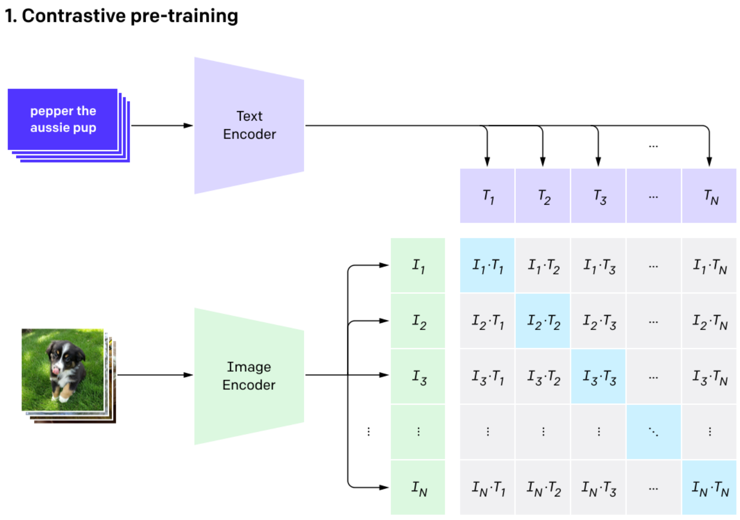
- We introduce Clip-DB, a framework encoding both images and text into a related embedding space, enabling semantic similarity comparisons based on vector distances.
- We demonstrate the implementation of this approach utilizing a vector database (Chroma DB) for efficient storage and retrieval of image embeddings.

2 Preliminaries

The following section provides a brief overview of the key technologies that form the foundation of our proposed semantic image search system: Contrastive Language-Image Pre-Training (CLIP) and Chroma DB. Understanding these components is essential for comprehending the methodology employed in this project.

2.1 Contrastive Language-Image Pre-Training (CLIP)

Contrastive Language-Image Pre-Training (CLIP) by Radford et al. (2021) is a neural network trained on a large dataset of images and their associated text descriptions. Developed by OpenAI, CLIP is designed to understand the relationship between images and text by learning to predict which text caption goes with which image in a batch. This is achieved through a contrastive learning objective that encourages the model to learn a shared embedding space where the embeddings of an image and its correct caption are close together, while embeddings of mismatched image-caption pairs are pushed further apart, shown in Figure 1



	T_1	T_2	T_3	...	T_N
I_1	$I_1 \cdot T_1$	$I_1 \cdot T_2$	$I_1 \cdot T_3$	...	$I_1 \cdot T_N$
I_2	$I_2 \cdot T_1$	$I_2 \cdot T_2$	$I_2 \cdot T_3$	...	$I_2 \cdot T_N$
I_3	$I_3 \cdot T_1$	$I_3 \cdot T_2$	$I_3 \cdot T_3$	...	$I_3 \cdot T_N$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
I_N	$I_N \cdot T_1$	$I_N \cdot T_2$	$I_N \cdot T_3$	...	$I_N \cdot T_N$

Figure 1: CLIP pre-trains an image encoder and a text encoder to predict which images were paired with which texts in our dataset.

The significance of CLIP lies in its ability to produce high-quality, multimodal embeddings. It consists of an image encoder and a text encoder that map their respective inputs into the same vector space. This shared embedding space allows for direct comparison of the semantic content of images and text using distance metrics like cosine similarity. Unlike traditional methods that rely on handcrafted features or keyword matching, CLIP captures richer, high-level semantic information. This capability makes it particularly well-suited for tasks requiring cross-modal understanding, such as text-to-image and image-to-image retrieval based on semantic content rather than just superficial features or metadata. In our project, CLIP serves as the core mechanism for generating semantic embeddings for both query inputs (text or image) and the images in our database.

2.2 Chroma DB

Chroma DB is an open-source vector database specifically designed for building AI applications with embeddings. It provides functionalities for storing, indexing, and searching high-dimensional vectors, which are the numerical representations of data produced by models like CLIP. Unlike traditional databases optimized for structured data or keyword-based search, vector databases are built to handle the unique challenges of working with embeddings, particularly the need for efficient similarity search over large datasets.

Key features of Chroma DB include its built-in support for similarity search algorithms (such as cosine similarity), an easy-to-use Python API for seamless integration into machine learning workflows, and the ability to store metadata associated with vectors. In the context of our semantic image search engine, Chroma DB serves as the backend for storing the pre-computed CLIP embeddings of the images in our dataset. When a query (either text or image) is encoded into an embedding using CLIP, this query embedding is then used to perform a similarity search within the Chroma DB to find the most semantically relevant image embeddings, and subsequently, the corresponding images. Its efficiency in handling vector similarity search is crucial for the performance of our system.

3 Architecture

Our proposed semantic image search system is designed as a unified framework capable of handling both text and image queries to retrieve semantically relevant images. The architecture is built around the principle of embedding images and text into a shared vector space, enabling cross-modal similarity comparisons. The overall architecture, as depicted in Figure 2, consists of several key components: Input, Encoders, Database, Matching, and Output.

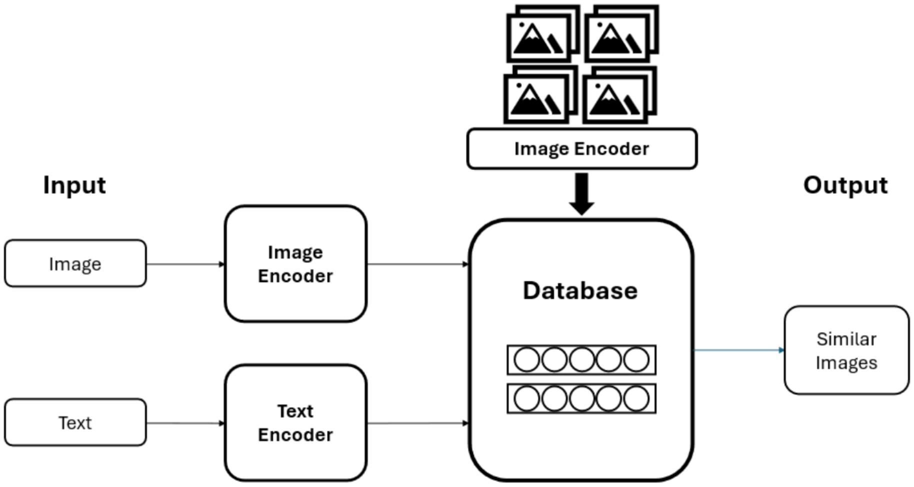


Figure 2: Architecture of the semantic image search system.

3.1 Input

The system accepts two types of queries: an image or a text string. This flexibility allows users to initiate a search using either a visual example or a textual description of what they are looking for.

3.2 Encoders

The core of our system relies on robust encoders capable of translating different modalities into a common representation space. We utilize the CLIP model, which provides both an Image Encoder and a Text Encoder.

- **Image Encoder:** This component processes input images and transforms them into high-dimensional image embeddings. These embeddings capture the semantic content of the images.
- **Text Encoder:** This component processes input text queries and transforms them into high-dimensional text embeddings within the same shared space as the image embeddings.

These encoders are crucial as they enable the comparison of images and text based on their semantic meaning, rather than superficial features or keywords.

3.3 Database

The system includes a database that stores a collection of images represented by their pre-computed embeddings. Prior to search operations, all images in the dataset are passed through the Image Encoder, and their resulting embeddings are stored. We employ Chroma DB as our vector database due to its efficiency in storing and querying high-dimensional vectors and its built-in support for similarity search.

3.4 Matching

When a query (either an image or text) is received and encoded into an embedding by the respective CLIP encoder, this query embedding is compared against the image embeddings stored in the database. The matching process identifies the images in the database that are most semantically similar to the query. To measure this semantic similarity in the shared embedding space, we utilize cosine similarity.

Our choice of cosine similarity is rooted in the nature of the CLIP embedding space. CLIP is trained to bring the embeddings of an image and its corresponding text description close together in this space, maximizing their cosine similarity. Therefore, if two different images (I_1 and I_2) are semantically related or depict similar concepts, it is expected that their embeddings (I_1 and I_2) will also be close to the embedding of a text description (T) that represents that common concept. As shown in the relationship $\cos(T, I_1) \approx 1$ and $\cos(T, I_2) \approx 1$ implying $\cos(I_1, I_2) \approx 1$, the high similarity between each image embedding and the shared text embedding suggests a high similarity between the two image embeddings themselves. Consequently, by calculating the cosine similarity between the query embedding and the stored image embeddings, we can effectively identify images that share similar semantic content, aligning with the human understanding of similarity.

3.5 Output

Based on the similarity scores obtained during the matching phase, the system retrieves and presents the images from the database that are most semantically similar to the input query. The results are typically ranked in descending order of similarity score, providing the user with the most relevant images first.

4 Results

4.1 Image Classification

Evaluating the performance of a semantic image search system based purely on subjective notions of ”similarity” can be challenging. To provide a more objective measure of our system’s ability to retrieve semantically related images, we transferred the retrieval task into a classification problem. This allows us to quantitatively assess whether the images retrieved by our system for a given input image belong to the same semantic category or class as the input.

For this evaluation, we used image datasets where each image is associated with a class label. The process involved taking an image from the dataset as a query, using our system to retrieve the most similar images from the rest of the dataset, and then checking if the class labels of the retrieved images match the class label of the query image. We measured performance using Top-1 and Top-5 accuracy, which represent the percentage of queries for which the correct class label was present in the top 1 and top 5 retrieved images, respectively.

We conducted this evaluation on the MNIST and CIFAR10 datasets in Table 1. The MNIST dataset consists of grayscale images of handwritten digits (0-9), while CIFAR10 consists of color images across 10 different classes (e.g., airplane, dog, cat). These datasets provide a good testbed for evaluating semantic understanding at a basic object level.

Table 1: Results of the classification-based evaluation on the MNIST and CIFAR10 dataset

Dataset	Top-1 Accuracy	Top-5 Accuracy
MNIST	93.8%	98.3%
CIFAR10	28.0%	66.5%

4.2 Tom chase Jerry

Beyond quantitative metrics, we also performed a qualitative evaluation to assess the system's ability to understand and retrieve images based on more complex semantic concepts, such as actions and characters within a scene. For this, we used an example that involved frames from the ”Tom and Jerry” animated series.

As input query, we used an image obtained from a Google search related to ”Tom chase Jerry”. Our database for this evaluation consisted of images created by sampling one frame every 5 seconds from three episodes of the ‘Tom and Jerry’ animation. These frames were processed through our CLIP encoder and their embeddings were stored in the Chroma DB.

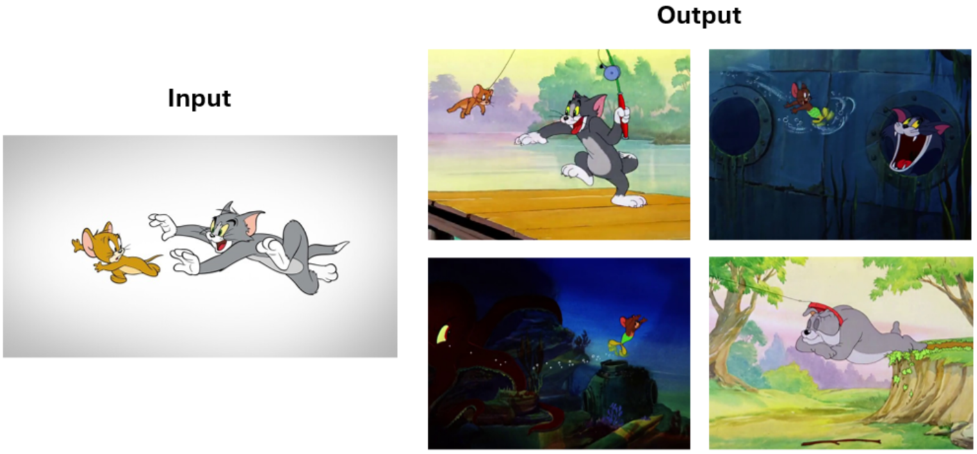


Figure 3: Image Search using ’Tom chase Jerry’ image

We then used the query image to perform a semantic search against this database of ”Tom and Jerry” frames in Figure 3. Upon examining the top retrieved images, we observed that among the first four results, three were semantically related to the action of ”chasing”. Furthermore, two of these images prominently featured the main characters, Tom and Jerry, engaged in a chase. This qualitative example demonstrates the system's capability to go beyond simple object recognition and retrieve images based on more dynamic and character-specific semantic content, aligning well with the query's implied meaning of Tom chasing Jerry.

5 Discussion

While our CLIP-based approach demonstrates significant promise in enabling semantic-aware image search, it is important to acknowledge some inherent complexities and limitations related to the definition and measurement of image similarity.

One key aspect is that the concept of "similarity" in images can be inherently vague and subjective. What one person considers similar to a query image might differ from another person's interpretation, depending on their focus. Humans may prioritize different aspects of an image, such as the main subject, the background, the overall scene, or even abstract concepts and emotions conveyed. The learned embedding space, while powerful in capturing semantic relationships from large-scale data, might not always perfectly align with the nuanced and diverse ways humans perceive and judge similarity. The single vector representation for an entire image is a simplification, and different aspects of the image contribute to this vector in ways that may not always isolate the most salient features according to human judgment for a specific search task, shown in Figure 4.

Furthermore, factors within an image that are not central to the desired semantic concept can potentially influence the generated embedding and, consequently, the similarity matching process. For instance, complex or cluttered backgrounds, variations in lighting, or the presence of irrelevant objects in the scene can introduce "noise" into the embeddings. While CLIP is designed to be robust, these extraneous elements might still affect the vector representation, potentially causing images that are semantically similar in their primary subject to be less similar in the embedding space due to differences in their backgrounds or other non-essential details. This can sometimes lead to retrieved results that, while having some shared features, might not be considered perfectly similar from a human perspective focused on the core semantic intent.

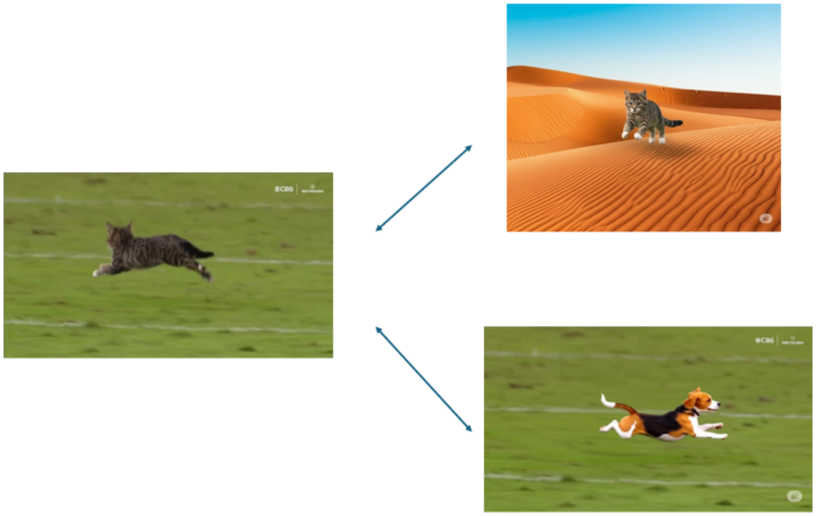


Figure 4: In an illustrative case, with a left image of a cat on grassland, a top-right image of a cat on sand, and a bottom-right image of a dog on grassland, our model finds the cosine distance between the left and bottom-right images to be smaller. This contrasts with a human assessment that prioritizes the animal subject, where the left image would be considered more similar to the top-right image.

To address these challenges and potentially improve the alignment between embedding-based similarity and human perception, future work could explore methods that allow for more explicit control over which parts of the image the model focuses on during the encoding process. By providing mechanisms to explicitly guide the model's attention to the semantically relevant regions of an image—for example, by indicating the main subject or areas of interest—it might be possible to generate embeddings that are less influenced by irrelevant or noisy content like the background. This could lead to more accurate and intuitively similar retrieval results, as the embeddings would more strongly represent the core semantic elements that are important for the search query. Approaches involving techniques like attention mechanisms or incorporating additional conditioning signals during training or inference could be explored in this direction.

6 Conclusion

In this report, we addressed the limitations of traditional image search methods in capturing semantic content by proposing a unified system leveraging Contrastive Language-Image Pre-Training (CLIP). By embedding both images and text into a shared semantic space and utilizing a vector database for efficient retrieval, our approach enables semantic similarity comparisons that align more closely with human understanding. Our experimental results, including performance on a classification task and a qualitative example, demonstrate the effectiveness of this semantic-aware retrieval method. While challenges related to the subjective nature of similarity and the influence of image composition exist, this work presents a robust framework for enhancing image search capabilities through deep semantic understanding.

References

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.