VISUAL SEARCH ENGINE USING VISUAL LANGUAGE MODEL

AI-Powered Image Retrieval Using Vision-Language Models

(Ravi kumar , Satjot singh , Ramandeep)

Mentor : Er Gursewak singh

2025

📌 Project Report Template (Problem 7)

🔍 Introduction

This project focuses on building a visual search engine using Vision-Language Models (VLMs). The system retrieves semantically and visually similar images based on text or image input, enabling efficient image search for large datasets. It has applications in e-commerce, digital asset management, and content-based retrieval systems.

🏗️ Project Overview

📌 Problem Statement

Develop a visual search engine that leverages vision-language models (VLMs) to retrieve relevant images based on textual queries or sample images. The system should embed both text and images into a shared representation space, allowing users to search via keywords, natural language descriptions, or example images.

🛠️ Technologies Used

List all major tools, libraries, and frameworks. For example:

- **Programming Language**: Python
- **Framework**: PyTorch / TensorFlow / Scikit-learn
- **Data Processing**: Pandas, NumPy, SciPy, PIL
- **Visualization**: Matplotlib, Seaborn
- **Web Framework** : Streamlit
- **Version Control**: Git & GitHub

🚀 Implementation

---

1 Data Collection / Preprocessing

🌐 **Source:** Publicly available at Flickr30K official website.

🖌️ **Cleaning or Transformation**

🧠 **Embedding Generation:**

- Extracted image embeddings using resNet18 vision encoder.
- Extracted text embeddings using resNet18 language encoder.

---

2) Feature Engineering or Model Building

✅ **Model Used:** Pretrained **CLIP (ViT-B/32)** from OpenAI.

🧠 **Function:** Converts images and text into a shared embedding space.

🏋️ **Training:** No full training; used **pretrained model** for embedding extraction.

🎯 **Image Embedding:** Extracted using CLIP's vision encoder.

📝 **Text Embedding:** Extracted from user queries using CLIP's text encoder.

🔍 **Similarity Search:** Performed using **cosine similarity** between embeddings.

---

3) Evaluation

🔍 **Recall@K**: Measures how often the correct result appears in the top **K** results (e.g., Recall@5 or Recall@10).

📈 **Precision@K**: Measures the proportion of relevant images among the top **K** retrieved.

🚫 **Confusion Matrix / ROC-AUC**: Not applicable here as this is not a classification problem.

---

4) Interface / Deployment (if applicable)

- Implemented a Streamlit web UI for interactive image selection and similarity search.
- Model loading & prediction

📂 Project Structure

Visual Search Engine using VLMs

|— 📂 data/ # Dataset

|— 📂 models/ # Saved model files

|— 📜 main.py # Main script

|— 📜 front_end.py # Web UI

|— 📜 requirements.txt # Dependencies

|— 📜 README.md # Documentation

🔧 Setup & Installation

1) Clone the Repository

```
git clone https://github.com/Krishnandu-Halder/Visual_Search_Engine_using_VLM.git
cd Visual_Search_Engine_using_VLM
```

2) Create and activate virtual environment

 # Windows

```
python -m venv venv
venv\Scripts\activate
```

 # Linux / macOS

```
python3 -m venv venv
source venv/bin/activate
```

3) Install dependencies

```
pip install -r requirements.txt
```

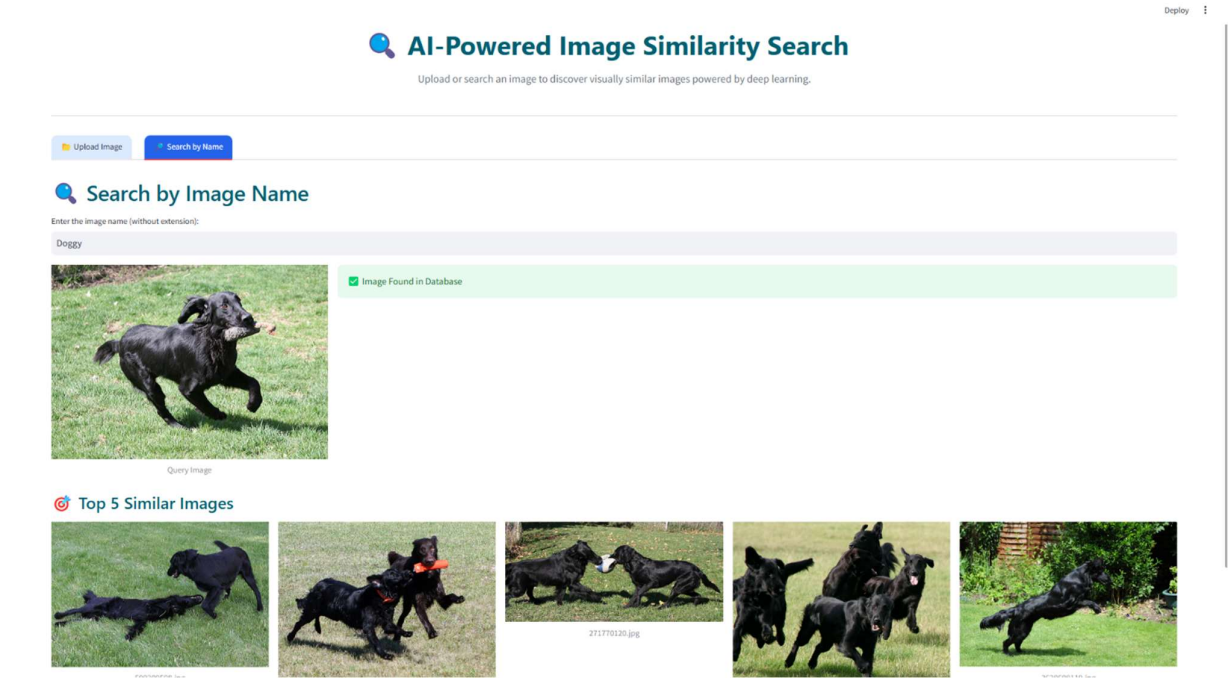
4) Run the Application

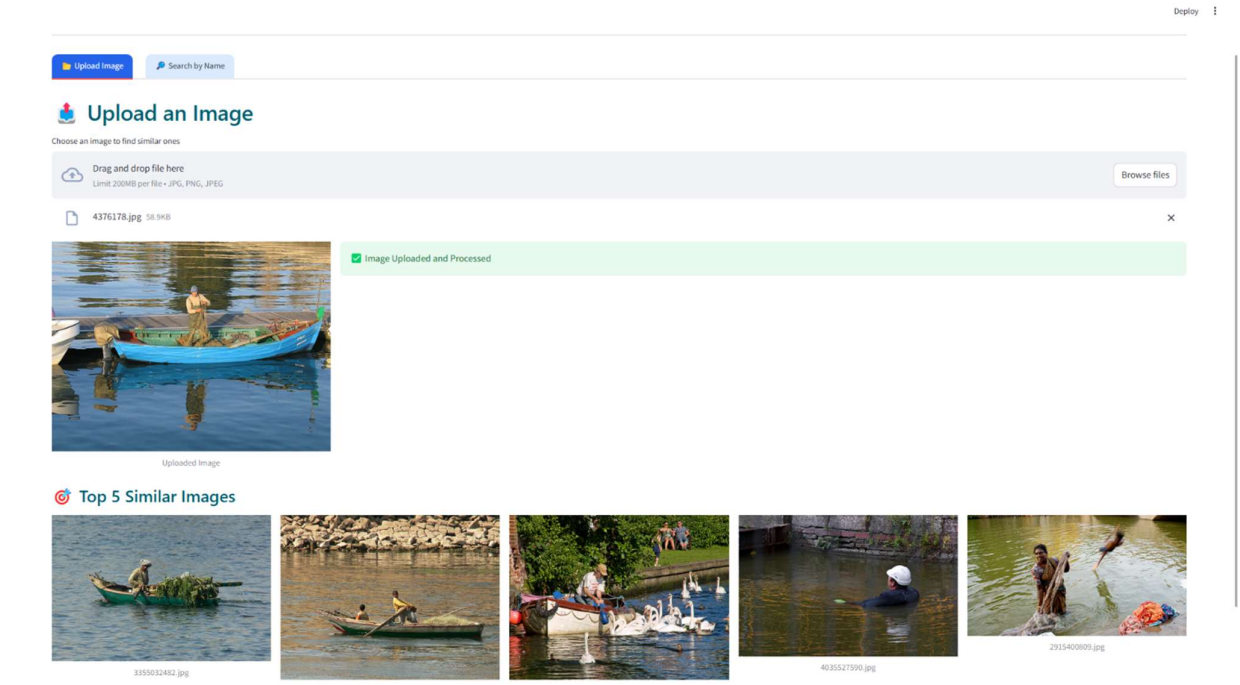
```
streamlit run app.py
```

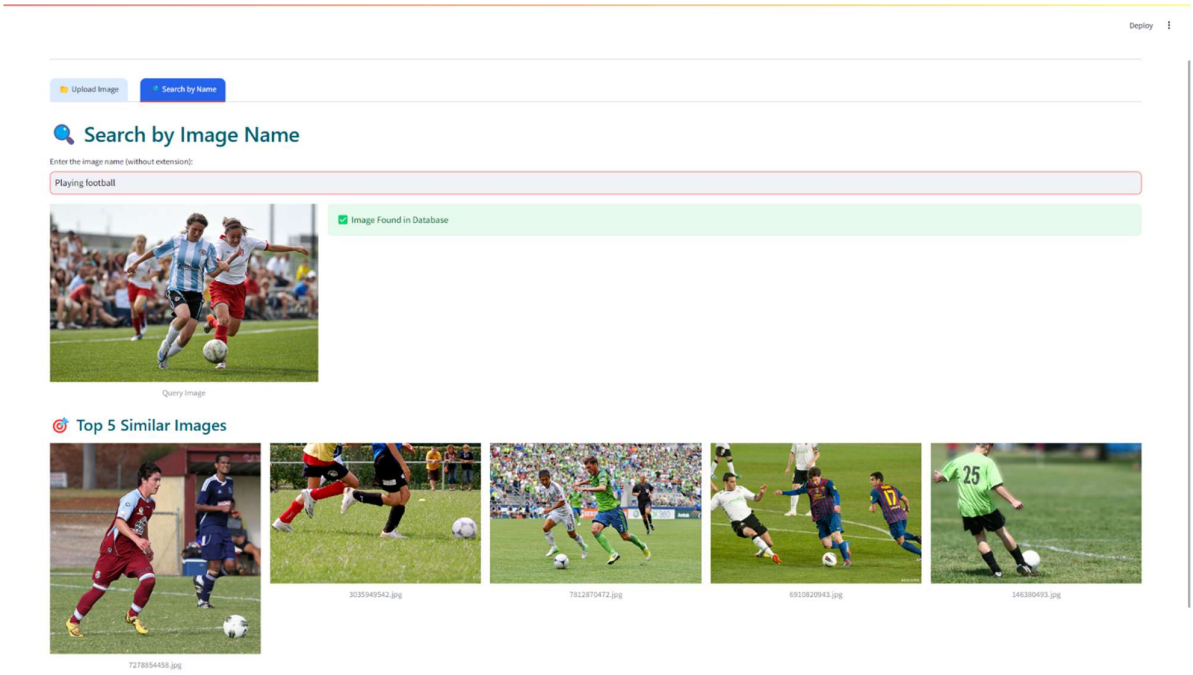
5) Deactivate

```
deactivate
```

Preview


Deploy
AI-Powered Image Similarity Search
Upload or search an image to discover visually similar images powered by deep learning.
Upload Image
Search by Name
Search by Image Name
Enter the image name (without extension):
Doggy
Image Found in Database
Query Image
Top 5 Similar Images
271770120.jpg


Deploy
Upload Image
Search by Name
Upload an Image
Choose an image to find similar ones
Drag and drop file here
Limit 200MB per file • JPG, PNG, JPEG
Browse files
4376178.jpg 58.9KB
Image Uploaded and Processed
Uploaded Image
Top 5 Similar Images
3355032482.jpg
4035527590.jpg
2915400809.jpg



📌 Future Improvements

- Integrate CLIP/DINO models for enhanced accuracy.
- Deploy to Cloud (AWS/GCP) for scalability.
- Optimize database storage for large-scale image retrieval.
- Add more interactivity in UI
- Use more advanced algorithms

Author-

Ravi kumar yadav - Frontend & AI Developer

For any queries, contact: kumaryadavravi016@gmail.com

🚀 **Thank you for exploring the Visual Search Engine!**