COMP 642 Final Report

Senthil Thanneermalai (st108), Kaushal Kumar Agarwal (ka62)

Reference: https://canvas.rice.edu/courses/64051/discussion_topics/374550

1. 1-2 paragraph description of the problem you have solved or explored.

1. Multimodal Image Retrieval Method: The project addresses the challenge of rapidly retrieving visually similar content from a vast dataset in response to multimodal queries. Traditional content retrieval systems often need help with the data scale and the input types' diversity. Our system innovates by implementing a retrieval method that processes image, text, and audio queries to find the most similar images from the SBU Captions Dataset, which contains over 1 million images. It took around 35 hours to extract all image features using the SigLIP (siglip-so400m-patch14-384) model. The system incorporates ANNOY (Approximate Nearest Neighbors Oh Yeah) and SOTA (State-Of-The-Art) USearch algorithms, ensuring fast and accurate similarity searches.

2. Comparison of ANNOY and SOTA USearch Algorithm: Retrieving k-similar vectors from a database containing around 1 million image vectors usually has high latency. This project focused on comparing latency and the accuracy of retrieving true positives while retrieving k-similar images using the abovementioned algorithms.

3. Object Detection and Search: Integrating YOLO (You Only Look Once) API for object detection further refines the search capability, allowing for object-specific queries. The YOLO API is leveraged to identify objects within an input image. The specific object the user chooses is searched within the image index to provide relevant results.

4. Multilingual Capabilities (Explored): The system's multilingual support extends its accessibility to a global user base. The model is designed with extensive multilingual support, capable of handling queries in over 100 languages. The extracted text features from queries in multiple languages can be directly compared in the Image index to deliver k-similar images. It has been observed that the current model does not support the Hindi language. This limitation can be addressed by fine-tuning the model using datasets like Hindi 8K Flickr or something more significant than that, thereby leaving scope to explore for future work.

2. 1-3 paragraph literature survey.

Authors in "Object-Based Image Retrieval From Database Using Combined Features" use Bi-directional Empirical Mode Decomposition (BEMD), Harris Corner Detector, and HSV Color Space to extract the feature vectors of objects in an input image. The system calculates the

similarity score between vectors by determining the histogram intersection and Euclidean distance between the input vector and the vectors in the database. This Brute-force approach will take more time and space to execute. We will use an approach known as Approximate Nearest Neighbor (ANN). This is very efficient in determining similar vectors. For our second use case of Object detection, we will use some SOTA models, such as YOLO, instead of manually calculating the edges using edge detectors.

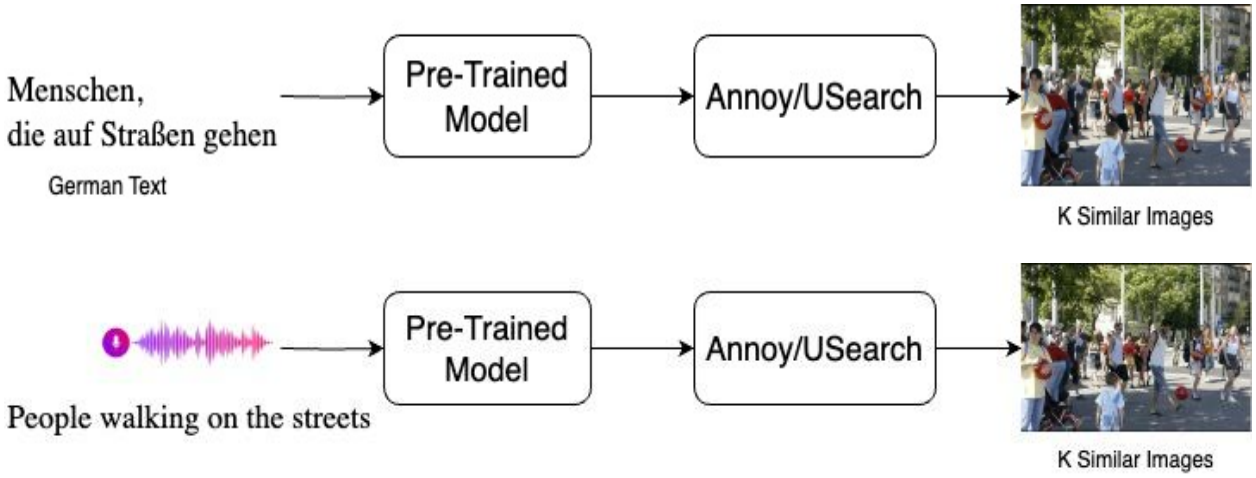
Authors in "Talk Don't Write: A Study of Direct Speech-Based Image Retrieval" directly convert the Speech to vector representations using two Encoders. The first encoder converts the spoken captions to high-dimensional feature vectors using WAV2VEC, and the second encoder uses EfficientNet-B4 to extract visual features. The Retrieval process is carried out by finding the dot product between all the feature vectors. The result is sorted in descending order to find the most similar image. Our system will use the SigLip model, which performs better in extracting feature vectors. Similarly, comparing feature vectors becomes very efficient when we use techniques like Approximate Nearest Neighbor to find similar vectors compared to the Brute force approach.

Authors in Clothing Retrieval Based on Image Bundled Features use a method known as SIFT (Scale-Invariant Feature Transform) features to extract local features in images. This is extremely useful for extracting feature vectors of clothes where patch patterns are essential in representing a certain type of cloth. They introduce a geometric constraint based on the SIFT distance matrix to group similar vectors. The overall similarity is calculated by aggregating the similarities of the bundled vectors. Incorporating a geometric constraint leads to a less efficient system due to increased latency in calculating similar vectors. Our system will use a SOTA object detector that converts the images to vectors with low latency.

3. One bold sentence about your hypothesis or argument for each use case.

A. **Image to Image, Text to Image, Audio to Image Retrieval:** We hypothesize that by using the ANNOY/USearch algorithms for image-to-image, Text-to-Image, and Audio-To-Image retrieval, we can efficiently retrieve k-similar images from a large dataset of 1 million image vectors extracted using the SigLip model. We argue that our approach of using the ANNOY and USearch algorithm for image retrieval is superior to traditional methods due to its ability to retrieve 1000 similar images from 1M index within 1.7 and 0.35 seconds, respectively, thereby balancing speed and accuracy.
B. **Object Detector combined with Image-Based Retrieval System:** We hypothesize that by using object detectors to extract objects from an image, we will be able to use the Image-based Retrieval system to recommend similar images of individual objects in an image based on user input that runs on vectors having size less than 1 GB in total size and able to predict results within few seconds/milliseconds of the search.

C. **Multilingual Text to Image Query:** The current system can remove the language barrier for querying k-similar images from an extensive database. The current output based on a Spanish, German, Italian, and French text query yielded good results for a 1 million index of image vectors. The SigLip model achieved zero-shot classification across more than a hundred languages with an accuracy of 84.5%.



4. 2-5 paragraphs describing the experimental settings for each use case.

Our experiments used the SBU Captions dataset, which contains 1 million images. With images preprocessed to a uniform size and format for extracting image features using the SigLIP model. The feature extraction was performed on a MacBook M2 Air with 16 GB RAM and only a CPU. The extracted features were stored in a CSV file for further processing.

We utilized Python's Annoy/USearch library to create an index for all the extracted feature vectors. This index was then used to find k-similar images for a given input image. The choice of k was varied to evaluate the performance of our system under different conditions.

To evaluate our results, we measured the latency of image-to-image use cases for both Annoy and USearch algorithms and evaluated the accuracy metric. This metric measures our system's ability to return the number of true positives in the top K results.

For **1M** Images (SBU Dataset):

Index creation time for Annoy: 3 mins 30 sec
Index creation time for USearch: 2 mins 20 sec

We used the SigLIP model to extract feature vectors from Images and Text. Once the feature vectors were extracted, we built a similarity index using these vectors to determine the similar vectors to that of the input query image/text.

We compared two vector similarity retrieval methods, Annoy and USearch, to determine the best one for our use. We see that USearch is faster and much more efficient than Annoy. Similarly, USearch's index creation time is faster than Annoy's.

Determining the ground truth variables is one major problem with determining the accuracy of the similarity index. When we query an input image, we can use our naked eyes to determine whether the images recommended by the index are similar to the query image. However, this task becomes unfeasible in case of 1000 or more images. Therefore, we use the YOLO object detector to determine the objects in the query image. Once this is determined, we run the object detector on the images recommended as similar by the index. Using this information, we calculate the accuracy score.

Fashion recommendation-
We used the DeepFashion-MultiModal dataset containing 44k images to recommend similar fashion apparel. We used a similarity search for K=10 with an input image given below and got the output as shown in the described picture below:

Input Image:

Output Images (10 similar images):

As we can see, the output shows images that are irrelevant to the input.
Note: We tried using multiple input images, but the output images were not found to be good.

Resolution Thought Process: We may need to change our process of extracting features from images. We will first need to detect the objects (jackets, shirts, shoes, etc.) from the image dataset (since it is combined with a human wearing it) using YOLO (You Only Look Once) API and then extract features of those objects using SigLIP. This may increase the chances of extracting similar images from the given dataset. We will try experimenting with this for the final report as well.

Fine-tuning SigLIP Multilingual model- An attempt has been made to fine-tune the Multilingual SigLIP model with only 500 images out of the 8K Hindi Flickr dataset due to compute and time constraints by minimizing the cosine similarity (difference) between the extracted image features and their respective caption (text) features. Unfortunately, the model got overfit with just four epochs, and the training loss became too negligible, such that the model prediction became static. Overall, the model outputted the same images for any input text (whether in Hindi, English, German, etc.).

Resolution Thought Process: The model needs to be fine-tuned with entire 8K images rather than doing it only for 500. In this way, the model can be saved from being overfitted and thus give better results. The learning rate, batch size, epochs, and other parameters can also be hyper-tuned to prevent the model from getting too much overfitted.

5. Plots and 1-2 paragraphs summarizing experimental results for each use case(Not Required for Mid Report but needs to describe the current status of experiments with details)

Similarity Index	Latency for retrieving K-similar Images (ms)			
	K = 10	K = 50	K = 100	K = 1000
ANNOY	11.99	198	223.54	1681.87
USearch	17.36	73.5	89.8	349.46

Fig: Latency comparison table for retrieving K-similar images

Image	Ground Truth	True Positives	Accuracy (%)
 Car	79	22	27.85
 Bird	66	29	43.94
 Chair	61	17	27.87
 Person	281	109	38.79

Fig: Similarity accuracy for ANNOY index (1000 images)

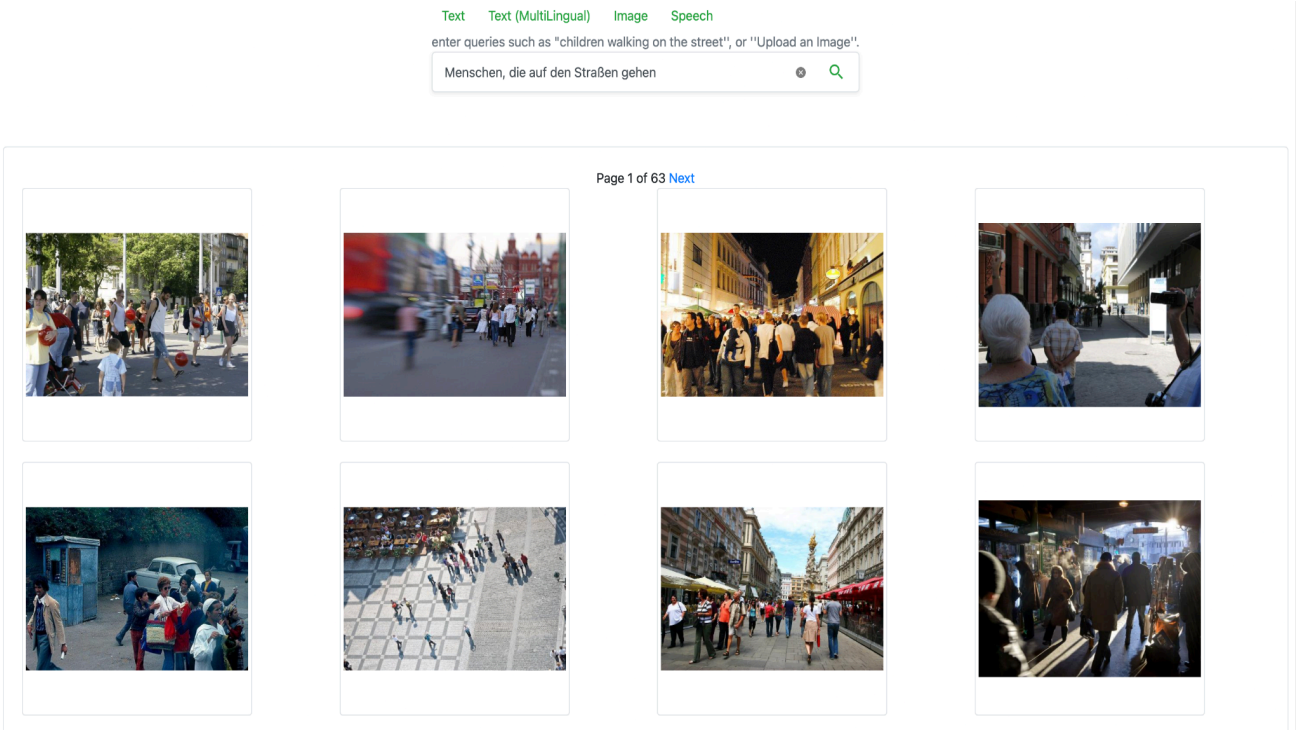
Image	Ground Truth	True Positives	Accuracy (%)
	**42**	**23**	**54.76**
	**43**	**9**	**20.93**
	**42**	**19**	**45.23**
	**14**	**2**	**14.28**

Fig: Similarity accuracy for USearch index (1000 images)

The attached tables above provide a summary of our current results. It shows the time taken to retrieve different numbers of similar images (10, 50, 100, 1000) for a given input image using both retrieval strategies. As can be seen from the table, our system can retrieve 1000 similar images from 1 million index of image vectors within a few seconds to milliseconds. We can conclude that USearch demonstrated greater efficiency, retrieving similar images nearly 5 times faster than ANNOY. In contrast, ANNOY and USearch fare similarly in retrieving similar images, with an average accuracy of 35%.

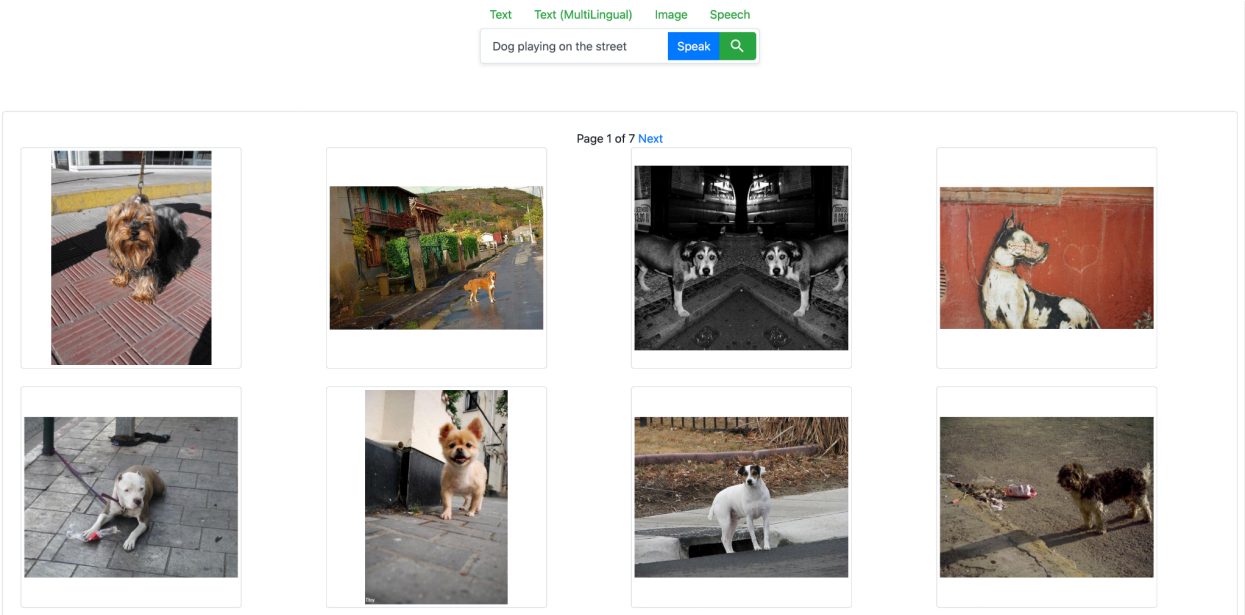
The above tables show that the accuracy score is low for both Annoy and USearch. The scores would have been better if we fine-tuned the YOLO object detector on our set of images before calculating the accuracy scores. This could prove beneficial in better object detection.

Multilingual results -



In the above image, we search for the German equivalent of “People walking on the streets.” The similar images retrieved by the USearch index accurately understand the multilingual search query text.

Speech-To-Image results-



In the above image, we speak the query “Dog playing on the street.” Similar images retrieved by the index accurately understand the speech query.

6. 1 paragraph conclusion of why you think you have justified 1 and 3. (Not Required for Mid Report)

- The platform accommodates diverse input types, such as Text-to-Image, Image-to-Image, Audio-to-Image, and Multilingual Text-to-Image search.
- YOLO object detection integration in the system enhanced the system's capabilities by incorporating an object inside an object similarity search. YOLO was also used to determine the ground truth labels for the similarity search output evaluation metric.
- Initiated fine-tuning the model on a subset of the Hindi language, which was previously unsupported.
- Further extension of this fine-tuning process to a larger dataset of Hindi language is planned.