DIP Project Mid Report

Project ID - 5
Project Title: Sketch Based Image Retrieval

Team Members:

- Santhoshini Gongidi (2018701020)
- Chris Andrew (2018701019)

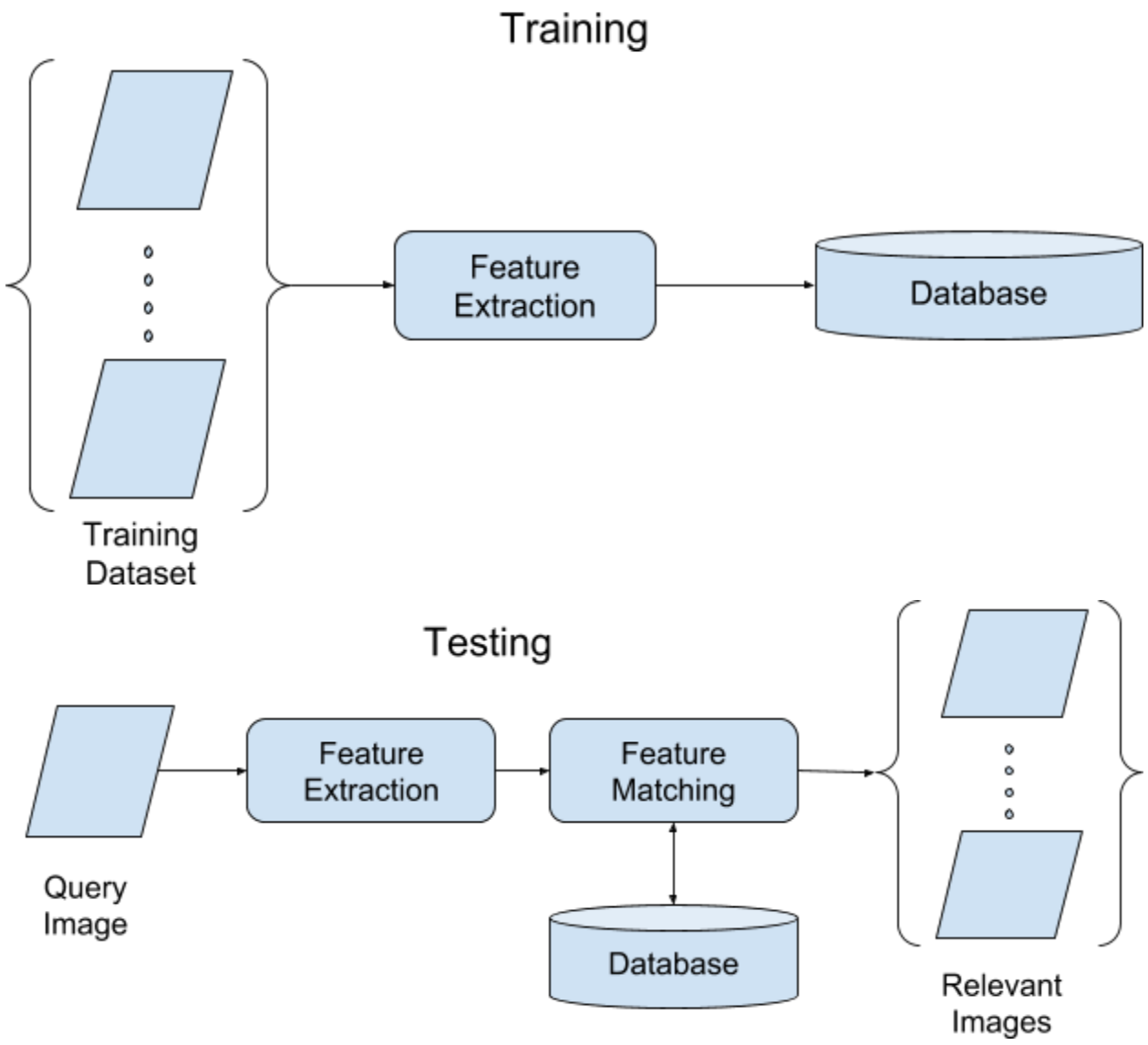
Github repository: https://github.com/Sanny26/sbir

Problem Definition:

In a **Sketch Based Image Retrieval(SBIR)** system, a given query sketch needs to be compared with images available in a database to retrieve images that are relevant to the query. This is a special type of **Content Based Image Retrieval(CBIR)** system, similar to the one used in Google's Image search(Search by Image), where an image is used to find similar images in Google's database.

The problem is not as simple as it seems, because of a large number of factors that introduce errors in the retrieval process. Digital/color images vary from sketches in that sketches only contain information about the edges. Comparing this edge information is not always easy and as such we must try and extract features from images that are invariant to color and extract information from the shape of the image. Apart from color, the images in the database will have a large variance in scale as well as the orientation of objects in the image. The features used to compare two images must also be scale and rotation invariant to allow a much more exhaustive search of the database.

Proposed Architecture:

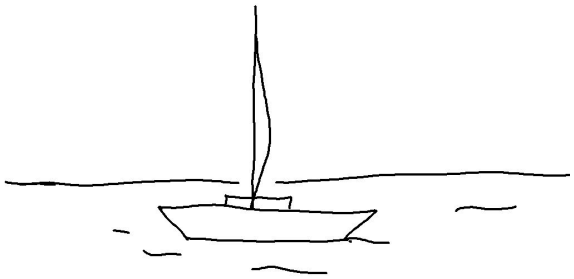


Dataset:

For our task, we will be using the benchmark dataset proposed in [1]. The dataset consists of sketches of 31 different object with 40 content images(photographs) for every sketch. The system is trained only using descriptors of the content images, and the retrieval is done only using information from the sketch images. This dataset has been carefully crafted for testing how well a SBIR system works and is widely used for benchmarking SBIR systems.

Sample from the dataset:

Content Image



Sketch Image

SBIR systems can be tested well even with small datasets as we only need to retrieve the top-k matching images. If these images are of the correct object then the system is said to perform well. Larger datasets require more computing time for matching images and are generally not used in test environments.

Method:

For the problem, we will be using a ***bag of features*** based model for feature extraction.

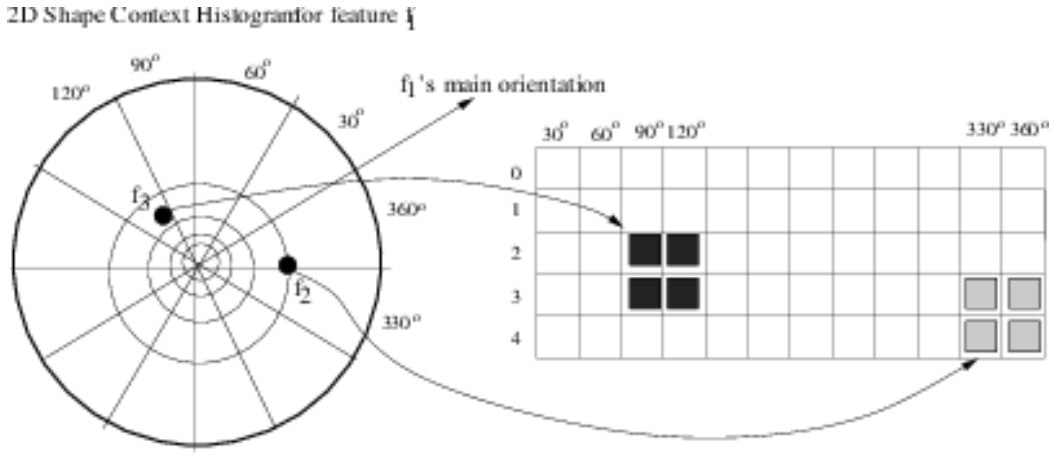
Most SBIR systems convert content images into the sketch space before comparing the features of the two for matching. SBIR systems have used prominent features for that have been used to match binary images, ex: SIFT, Harris corner, Edge distributions, Gabor, GSCM, etc.

1. In the method we use, we first convert the content image to a sketch using Canny Edge detection to detect the edges. These images are then used to extract features that are matched with the features of the sketch image.
2. For the bag of features, we first sample 500 random points on the edges from a content image. We then find descriptors around these 500 samples points. We concatenate the various types of descriptors to get a large descriptor for each sample point.
3. We then do KMeans clustering on these points and their descriptors, extracted from all images in the training set(content images). The points clustered together constitute a visual word in the sketch space. We try and find a set of 750 words from these points, by setting the number of clusters in the KMeans model. The cluster centers are saved so that they can be used at the time of testing.
4. For each content based image, we then compute a histogram of visual words, which will be our ***bag of features*** descriptor. This feature is used to match the sketch and the content based images.
5. During testing, we sample 500 points from the test image and extract multiple descriptors similar to what we did before. We then match the points to the previously obtained visual words using the pre-trained KMeans model and then find the distribution of the words on the sketch images to get a ***bag of features*** descriptor.
6. We then use our feature matching technique to match the sketch image descriptor to the closest content images having similar descriptors.

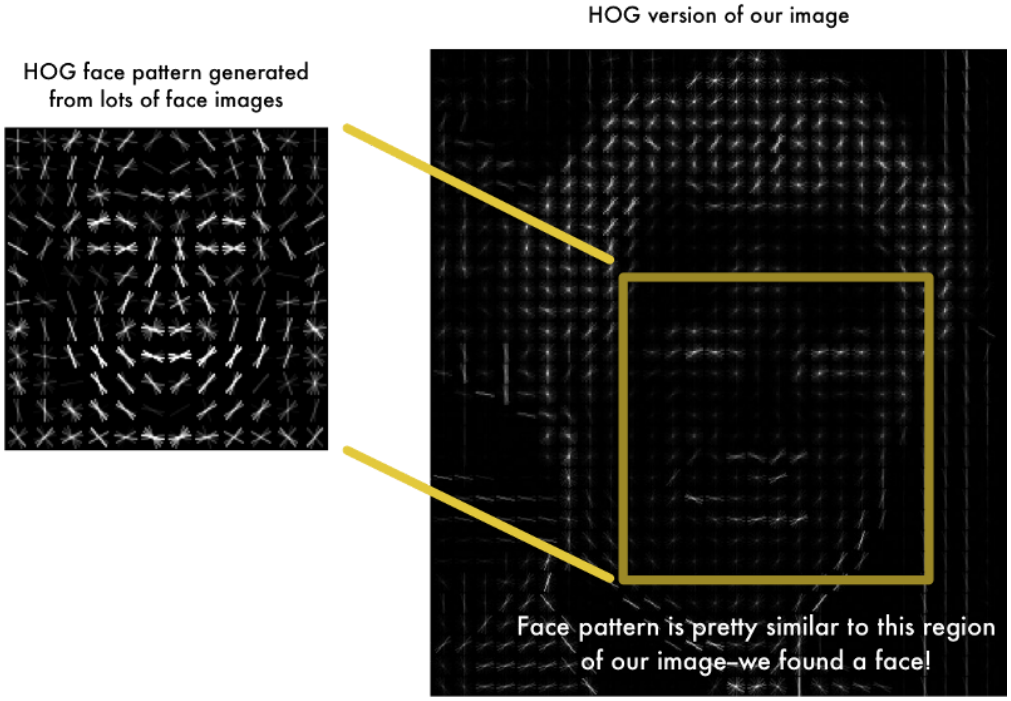
Descriptors for training bag of features model:
For each of the sampled points, we find the Shape Context[2] features, Histogram of Oriented Gradients[3], SPARK[1] and SHoG[1] features around a region surrounding the point as proposed in [1]. These features are then concatenated together to form one large descriptor for the point, which is then clustered to get a visual word.

Features:

Shape Context Features: Shape context features encode the distribution of sample point locations on the shape relative to each of the other sample points.



Histogram of Oriented Gradients: Histograms of oriented gradients (HoG) are 3D histograms encoding the distribution of gradient orientations in small local areas. They were used primarily for face detection. The variant used in [1] is similar to that of the SIFT descriptor [4x4 windows with 8 directions].



SPARK descriptors: SPARK is an extension of shape context features. The main difference being the way sample points on the sketch are generated as well as the local region they describe.

SHoG: Structured Histogram of Oriented Gradients improve the standard histogram of oriented gradients descriptor by storing only the most dominant sketched feature lines in the histogram of oriented gradients. This helps to make descriptors extracted from user sketches and descriptors extracted from the database image more similar, improving the probability that descriptors that are near in feature space correspond to perceptually good matches.

Feature Matching:

Once the ***bag of features*** are generated for each of the content images, they are stored in the database for retrieval. Retrieval of the content based images is done using the popular Tf-iDF algorithm [4] for information retrieval. Tf-iDF, short for term frequency–inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. This can be used for the visual words in in both the content image as well as the sketch image to find the best matches for the given sketch image.

The method used in [1] defines a rank correlation based method to retrieve the best image, however we feel that given a bag of visual words is available to us, we can get better performance using the state of the art in information retrieval. This still needs to be tested out and we will try both rank correlation as well as Tf-iDF based methods.

References:

[1] Eitz, Mathias, et al. "Sketch-based image retrieval: Benchmark and bag-of-features descriptors." *IEEE transactions on visualization and computer graphics* 17.11 (2011): 1624-1636.

[2] Belongie, Serge, Jitendra Malik, and Jan Puzicha. *Shape matching and object recognition using shape contexts*. CALIFORNIA UNIV SAN DIEGO LA JOLLA DEPT OF COMPUTER SCIENCE AND ENGINEERING, 2002.

[3] Dalal, Navneet, and Bill Triggs. "Histograms of oriented gradients for human detection." *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 1. IEEE, 2005.

[4] Leskovec, Jure, Anand Rajaraman, and Jeffrey David Ullman. *Mining of massive datasets*. Cambridge university press, 2014.

Milestones:

Task	Pending/Completed/Partially
Review of the existing methods for SBIR and CBIR.	Completed
Collection of data for the SBIR system(Also includes pre-processing for noise removal/data augmentation)	Completed
Finalising the feature extraction mechanism and system design.	Completed
Implementation of the feature extraction mechanism.	Partially
Incorporate possible improvements and new techniques to improve the quality of the extracted features.	Partially
Implementation of the feature storage and Image retrieval systems.	Pending
Implementation of the feature matching mechanism.	Pending
Experiments to measure the performance of the SBIR.	Pending
Implementation of the search UI for uploading images to search in the database.	Pending