CSE 515: Multimedia and Web Databases

Title

CSE 515 Fall 19 Phase 1

Submitted by Manoj Tiwaskar (1216281170)

Group Members

Athul Pramod

Anjali

Prashant Singh Chauhan

Md Shadab

Tyler Giles

Manoj Tiwaskar

Abstract

The goal of phase 1 of the project is to get familiar with the 11k hand's image dataset and understand the concept of feature extraction and implementation of the two image feature extraction models. The two implemented models are Scale-invariant feature transformation and Local binary patterns, which helped in understanding the feature vector space/ feature descriptor of each implemented model. Once the features are extracted for the images, these features are used to retrieve the k similar images from the database for the given image. To find k similar images, the similarity/ distance functions are implemented, and the results obtained using these functions showed the importance of each function and characteristics of the images it captures.

Keywords: Histogram of Oriented Gradient, Local binary patterns, Scale-invariant feature transformation, Color Moment, Image Gradient, Feature Descriptor.

1. Introduction

Image Features is the information related to the image which can help for solving the computational task related to the application like detecting the objects in the scene, points, and edges. The process of finding the local neighborhood operations applied to an image is referred to as **feature Extraction**. The distinction in the images becomes relevant when the resulting detected features are relatively sparse. There are many features extraction models present like Color Moment (CM), Histogram of Oriented Gradient (HOG), Local binary pattern (LBP), Scale-Invariant Feature Transform (SIFT) and many more such feature extraction. Each of the feature extraction models detects different aspects of the image. In this phase, I have implemented SIFT and LBP.

Scale-Invariant Feature Transform (SIFT)

SIFT is the algorithm used for **image alignment and object recognition**. The algorithm is robust to detect features at different scales, angles, and the illumination of the image. It detects the interest points in the image, and the characteristics of the image are stored in the data structure called **Descriptor**. These extracted descriptors are used to identify the object in the training image set when attempting to locate the object of the test image. To perform reliable recognition, it is important that the features extracted from the training image be detectable even under changes in image scale, noise, and illumination. Such points usually lie on high-contrast regions of the image, such as object edges [9]. There are four steps involved in the SIFT algorithm:

1. Scale-space Extrema Detection:
   In this step, the key point is to identify locations and scales that can be repeatably assigned under differing views of the same object. Detecting locations that are invariant to scale change of the image can be accomplished by searching for stable features across all possible scales, using a continuous function of scale known as scale space.

2. Keypoint Localization:
   In this step, the keypoint locations are found; they have to be refined to get a more accurate result. They used Taylor series expansion of scale-space to get a more accurate location of extrema, and if the intensity at these extrema is less than a threshold value, it is rejected.

3. Orientation Assignment:
   In this step, the orientation is assigned to each keypoint to achieve invariance to image rotation. A neighborhood is taken around the keypoint location depending on the scale, and the gradient magnitude and direction is calculated in that region. An orientation histogram with 36 bins covering 360 degrees is created. The highest peak in the histogram is taken, and any peak above 80% of it is also considered to calculate the orientation. It creates keypoints with same location and scale, but different directions. It contributes to the stability of matching.

4. Keypoint Descriptor
   Now keypoint descriptor is created. A 16x16 neighborhood around the keypoint is considered. It is divided into 16 sub-blocks of 4x4 size. For each sub-block, eight bin orientation histograms is created. So, a total of 128 bin values are available. It is represented as a vector to form keypoint descriptor.

Local Binary Pattern (LBP)

LBP algorithm is a powerful feature for texture classification. It labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number. The most important property of the LBP operator in real-world applications is its robustness to monotonic gray-scale changes caused, for example, by illumination variation [8]. The feature vector of LBP is computed in the following way:

1. Divide the pixel in the window of 8x8 pixel.
2. For each pixel in the image is compared with its eight neighbor pixels. Then follow the pixels along a circle, i.e., clockwise or counterclockwise.
3. Where the center pixel's value is higher than the neighbor's value, we write 0. Otherwise, write 1. This gives an 8-digit binary number which is converted to decimal.
4. The histogram is computed over the cell, of the frequency of each number occurring (i.e., each combination of which pixels are smaller, and which are greater than the center). This histogram is of a 256-dimensional feature vector.
5. histograms of all cells are concatenated. This gives the feature vector for the entire window.
6. The feature vector of all the windows are computed, and then those are concatenated to get the feature matrix.

1.1 Terminology

1. Image Gradient: it is the directional change in the intensity or color in an image.
2. Histogram of Oriented Gradient: it is the feature descriptor used in image processing for detecting the object in the image.
3. Scale-Invariant: Changing the scale by a certain amount does not change statistic's shape or properties of the image.
4. Cosine Similarity: "It is a measure of similarity between two non-zero vectors of an inner product space that measures the cosine of the angle between them" [7].

$$\text{Cos}(\Theta) = \frac{A.B}{||A||\ ||B||}$$

5. Euclidean Distance: It is the square root of the sum of squared differences between corresponding elements of the two vectors.

$$D(X, Y) = \sqrt{\sum_i^n (Xi - Yi)^2}$$

1.2 Goal Description

The goal of this phase is to understand the concept of image features and the feature vector space. To extract the feature vectors from the images, I am implementing the two feature extraction models out of four those are SIFT and LBP. Using these feature vector obtained from the different model, we need to find the K similar images in the database for the given query image. The whole project of phase 1 is divided into four tasks:

1. Getting familiar with the dataset, which consists of 11k hands images used by Dr. Mohmoud Afifi for his research “Gender recognition and biometric identification using a large dataset of hand image.”.
2. I am implementing a program to extract feature using two feature extraction model (out of four given model, i.e., Color Moment, Local Binary pattern, Histogram of Oriented gradient and Scale-invariant feature transformation) which return unified feature descriptor for given image ID and model name.
3. Implementing a program for extract and store the feature descriptor of all the images in the given folder and model name.
4. Implementing a program to return k similar image for the given query image, model name to be used to find the similarity using its feature vector and k size.

1.3 Assumption

1. In the sift implementation, the number of key points (i.e., 70), which has been selected based on the response size of key point, is sufficient for finding the k similar images in the database. The different sets of keypoints have tried out before coming to the conclusion and the 70 keypoints gave promising results.
2. All the images that are processed at this stage are of the same size (1600x1200 pixels) and have an identical white background.
3. The file system space will be sufficient to store the feature descriptors of all the sample images for the given feature extraction model.

2. Proposed Solution/Implementation

SIFT feature extraction model:

The sift feature extraction is done using python OpenCV library name CV2. The image is first converted to a grayscale image. Which is then used to find the key points in the image. Then for the identified key points, the feature descriptor is computed. The number of keypoints varies for different images, and it was found that every image has 70+ key points. I have decided to sample only 70 keypoints which has high response value. The response defines how strong the key point is. The higher the value, the more likely the feature will be recognized among several instances of an object. For those sampled keypoints, the feature descriptor is computed.

To find the k similar image for the given query, I have implemented the algorithm which finds the best matching key points for each feature descriptor of query images into feature descriptors of the image in the database using Euclidean distance. Then the average of all the Euclidean distance for best keypoint match is calculated. The calculated value shows the similarity score of the query image with the image in the database. Lower the similarity score, the similar the two images are.

LBP feature extraction model:

In the LBP feature extraction model, the image is first converted from the RGB color model to the Grayscale color model. This is done using the OpenCV library. The above-described process is repeated on 100x100-pixel blocks for each block of the image. In computing the LBP for each block, the scikit-image library is used. To do this a radius of 1 is selected while the number of points in the immediate neighborhood of the block to analyze was set to 8. Selecting these values allows the program to get the LBP for the immediate neighborhood of each block. The LBP for each block of the image is computed using local binary pattern function from scikit-image library which took the parameter set to the uniform. Then the histogram is created of the LBP extracted data into ten bins with 11 edges, and each bin is having a range going from 0 to 10. This resulted again in 192 lists of histogram data separated into ten bins for each block of the image. By appending each of these lists of histogram data, a unified feature descriptor for each image that was processed was then created.

To find the k similar images for the given query image, the cosine similarity is used to find the similarity between the feature descriptor of the query image and the images in the databases.

3. Interface Specifications

3.1 Query specification

Queries can be passed as a command-line argument as various flags have been implemented in main script as:

3.2 Displaying results format

The results of top similarity score are displayed on the console for the given image where it displays the list of k most matched images along with their respective scores. Also, a directory will be created under output folder which will contain the output images. The feature descriptor are stored in xlsx file under the specific folder for LBP and sift model.

4. System requirements/ installation

- The project is implemented using Python 3.7, and following are the libraries that have been installed using pip utility in the command prompt:
    1. pip install OpenCV
    2. pip install Opencv-contrib-python
    3. pip install skimage
    4. pip install numpy
    5. pip install pandas
- OS version: Windows 10/8/7 x64 or macOS 10.8 or later or GNOME or KDE desktop.
- Memory: 4 GB RAM
- Storage: 1 GB disk space is sufficient to store the images and feature descriptors.

5. Execution Instruction

The project can be executed by running the script in the following manner in the command prompt:
Python Start_Script.py -m <Model_name> -d <Input_Directory_name> -i <Query_Image_name> -k < k value> -r

Importance of parameters:
-m : it is used to pass model name for which the features descriptor are extracted / to use the feature vectors to that model to find k similar images.
-d : It is used to pass the input directory of images.
-i : It is used to pass the query image name.
-k: It is used to pass the count of similar images that has to be retrieved for the given query image.
-r: It is used to rank the retrieved images and print their similarity score.

6. Related Work

In the paper [5], the sift keypoints are powerful features used mainly due to its distinctiveness. These features are robust across a substantial range of distortion, noise, rotation, scale, and illumination. The similarity of two images has been found out using best candidate match for each keypoint. The best candidate match for the given image is defined as the keypoint with minimum Euclidean distance for the invariant feature descriptor vector.

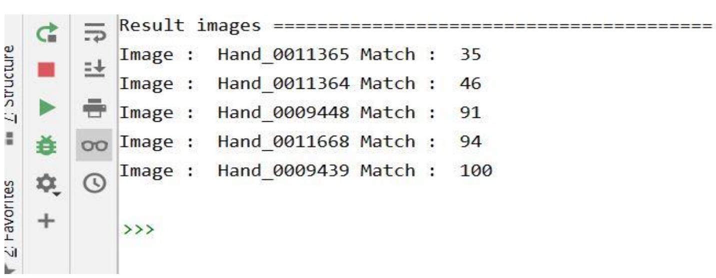
As mentioned in [1] and [2] LBP feature descriptors have proven to be valuable in applications involving face image analysis and image retrieval. This feature descriptor can give a summarized view of the structures within images by comparing each pixel with its neighboring pixels in the image. In selecting a similarity measure for LBP feature descriptors, it was found that cosine similarity has had success in other applications as stated in [2]. For this stage of the project, the cosine similarity function was then selected to determine the similarity of the images in the 11k Hands Dataset.

7. Result

1. Sift results: Following are the results of k=5 similar image using sift feature descriptor.

Query Image: hand_0011362.jpg

The Outputs are:



Hand_0011365.jpg Hand_0011364.jpg Hand_0009448.jpg

Hand_0011668.jpg Hand_0009439.jpg

2. LBP results: Following are the results of k=5 similar image using sift feature descriptor.

Query Image: Hand_0008662.jpg

The outputs are:

```
The top  5 similar images are:
('Hand_0008662', 0.0)
('Hand_0008663', 20064.331699951344)
('Hand_0008664', 22611.814728719703)
('Hand_0009001', 150322.30993327306)
('Hand_0008130', 152433.20944581187)
```

Hand_0008662.jpg

Hand_0008663.jpg

Hand_0008664.jpg

Hand_0009001.jpg

Hand_008130.jpg

8. Conclusion

The project helped in understanding the concepts of image feature extraction model and the feature descriptor of each model. Then these feature descriptors of each model are used to retrieve the k similar images. The results of k similar images using sift model tells us that sift detects the edges and return the results in the context of orientation of hand in the given image using Euclidean distance. The results of k similar images using LBP return the top 3 images out of top 5 was accurately based on the texture in the image. The actual database has only 3 images of that orientation, so the remaining results are retrieved based on the similar texture of top 3 results. For future phases of this project a conjunction of distance measurements on different feature descriptors may prove to be more useful to find the most similar images to a given image instead of using just one type of feature descriptor.

Bibliography

[1] D. Huang, C. Shan, M. Ardabilian, Y. Wang and L. Chen, "Local Binary Patterns and Its Application to Facial Image Analysis: A Survey," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 41, no. 6, pp. 765-781, Nov. 2011.

[2] Q. Cao, Y. Ying and P. Li, "Similarity Metric Learning for Face Recognition," 2013 IEEE International Conference on Computer Vision, Sydney, NSW, 2013, pp. 2408-2415

[3] Local binary patterns. (2019, August 25). Retrieved from https://en.wikipedia.org/wiki/Local_binary_patterns

[4] Mahmoud Afifi. "11K Hands: Gender recognition and biometric identification using a large dataset of hand images." M. Multimed Tools Appl (2019) 78: 20835.

[5] David G. Lowe "Distinctive Image Features from Scale-Invariant Keypoints" 2004 International journal od Computer Vision 60(2):91–110, Jan. 2004.

[6] SIFT - Scale-Invariant Feature Transform. (2019, September 9) Retrieved from http://weitz.de/sift/

[7] Cosine similarity - Wikipedia. https://en.wikipedia.org/wiki/Cosine_similarity

[8] Local Binary Patterns - Scholarpedia. http://scholarpedia.org/article/Local_Binary_Patterns

[9] Learn about various Object Detection Techniques. https://iq.opengenus.org/object-detection-techniques/

Appendix

Specific role of team members

Please see below how the models were divided among the team:

- Md Shadab: Local Binary Patterns and Histogram of Oriented Gradients
- Prashant Singh Chauhan: Color Moments and SIFT
- Tyler Giles: Color Moments and Local Binary Patterns
- Anjali: Color Moments and Histogram of Oriented Gradients
- Athul Pramod: Histogram of Oriented Gradients and SIFT
- Manoj Tiwaskar: Local Binary Patterns and SIFT