Subject: Pattern Recognition
Final Project
Lecturers: Prof. Dr. Sayed Fadel & Dr. Safwat Hamad
TAs: Mohamed Gawish & Islam Ibrahim

Ain Shams University
Faculty of Computer and Info. Sciences
Computer Science Department
1st Semester 2017/18

Final Project Title

Object Detection and Recognition

1. Introduction

An essential part of the behavior of humans is their ability to recognize objects. Humans are able to recognize large numbers of other humans, letters, digits, and so on.

The object recognition problem can be defined as a labeling problem based on models of known objects. Formally, given an image containing one or more objects of interest (and background) and a set of labels corresponding to a set of models known to the system, the system should assign correct labels to regions, or a set of regions, in the image.

2. Objective

The goal of this project is to build an object recognition system that can pick out and identify objects from an inputted camera image, as shown in Figure 1, based on the registered objects.

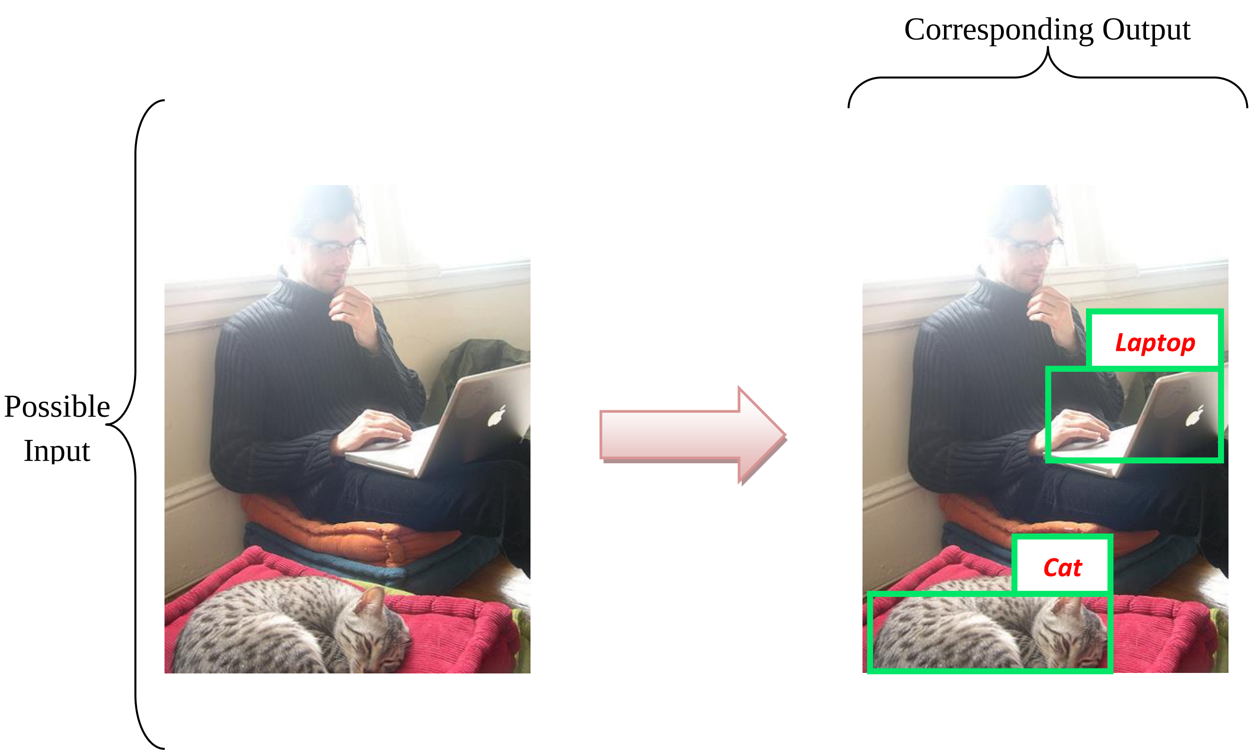

Figure 1. Desired performance of the project.

3. System Architecture

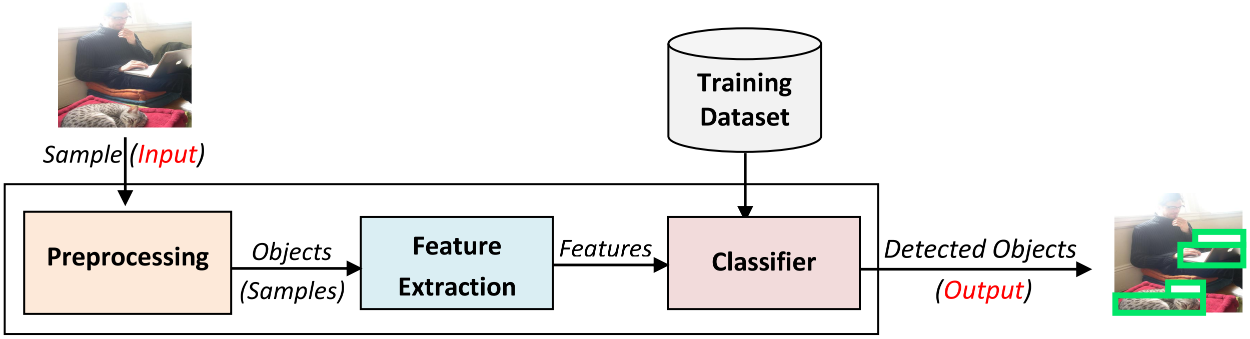

Figure 2. System Architecture

4. Dataset

- The dataset is real-world data, gathered from ImageNet [1]. **ImageNet** is an image database organized according to the WordNet hierarchy [1].
- The training dataset has 5 objects (classes) as shown in Figure 3. Five different models each.
- The test dataset has 14 different images, each image may contain one or more objects as shown in Figure 4.
- Each training model/testing image is given as colored image in .jpg format.

Figure 3. Objects (Cat, Laptop, Apple, Car, and Helicopter) [1]

Figure 4. A test image contains Helicopter, Cat, and Apple.

5. Preprocessing

- Perform any useful preprocessing routines on images, such as converting them to grayscale.
- As an image, contains more than one object, can be entered into the system. It is required to isolate objects from the background. For that, implement any proper segmentation method that can
    a. Detect objects from the preprocessed images.
    b. Crop the detected objects from the segmented images.
- Cropped objects are the samples to be classified in the next steps.

6. Feature Extraction

Features will be extracted of each object through using:

a. The gray-level co-occurance matrix (GLCM) to extract 4 features (Contrast, Entropy, Energy, and Homogeneity)
   i. The spatial relations are (1,0), (0,1) and (1,1).
   ii. The size of matrix is 6*6 (i.e. scale each image to gray level integer values between 0 and 5).
   iii. The matrix should be normalized and symmetric.
   iv. The final value per feature is the average of the three values obtained from the three spatial relations.

b. The run-length matrix to extract 11 features, as mentioned in the lecture
   i. The four directions (0, 45, 90, 135).
   ii. Scale each image to gray level integer values between 0 and 5.
   iii. The final value per feature is the average of the four values obtained from the four directions.

7. Classification Algorithms

Implement the following three classifiers for recognizing objects in **ONE** package:

No.	***Classifiers***	***Notes***
1	Modified ***K***-Nearest Neighbors	Choose the best ***k***
2	***r***-Near Neighbors	Choose the best ***r***
3	Support Vector Machine	Multi-Classification

8. Ouput

- Identifying the detected objects on the image by drawing a rectangle around each object on the image, as shown in Figure 5.
  - For drawing the rectangle, use the location of objects. So, you need to find the minimum X and Y, and the maximium X and Y. Minimum X and Y represent the lower-left point, (maximum X – minimum X) is the rectangle width and (maximum Y – minimum Y) is the rectangle height.

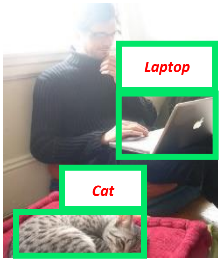

Figure 5. Output.

9. System Performance Evaluation

Separately, evaluate the performance of the three classifiers by the following techniques:

a. Confusion Matrix

use it to describe the performance of a classifier on *test images*, by representing actual and predicted classifications done by that classifier.

b. Overall Accuracy

use it to obtain the accuracy of a classifier from confusion matrix, by counting the number of *correct classifications*, and then dividing this by the total number of *classifications*.

10. Requirements

a. The user must be able to insert an input (image) to the application, and the application has to identify objects on the inputted image.
b. The user should be able to choose the feature extraction algorithm and the classification algorithm to be used with the current run of the system.
c. Using the test images you will test your classifier. Find the performance of your classifier using the Overall Accuracy (OA) and Confusion Matrix.
d. Display the validation curve of the first two classifiers.
e. Deliver a comparative study showing the effect of extracting features using
    a. GLCM only,
    b. Run-Length Matrix only, and
    c. GLCM + Run-Length Matrix

  on classifiers performance based on the two evaluation measures mentioned above. Thus, a report template will be provided for filling it.

11. Bonus

- Use Scale Invariant Feature Transform (SIFT) algorithm [2] to extract features of a colored image.
- SIFT describes image features that have many properties that make them suitable for matching differing images of an object or scene.
- The features are invariant to image scaling and rotation, and partially invariant to change in illumination and 3D camera viewpoint.
- An important aspect of this approach is that it generates large numbers of features that densely cover the image over the full range of scales and locations.
- As shown in Figure 6, given an image, SIFT generates a set of keypoints, each keypoint consists of its location, scale, orientation, and a set of 128 descriptors.

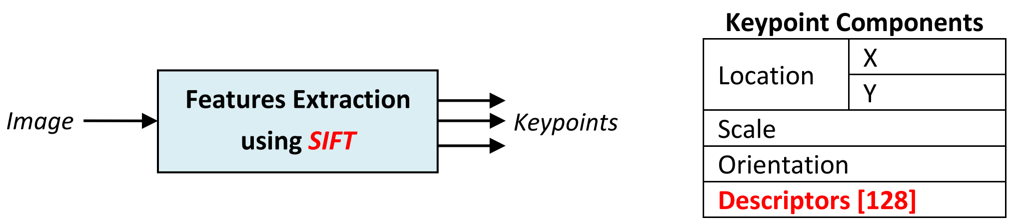

Figure 6. Features Extraction using SIFT

- Using SIFT, keypoints are the samples, and their features are the 128-element feature vector (descriptors) for each keypoint.
- **In testing phase only**, after classifying each keypoint of the inputted test image, count the number of keypoints that belong to each class. If the number of adjacent keypoints that belong to class(***i***) is greater than **10**, then the object of class(***i***) exists in the image.
- You can use the implementation of SIFT in VLFeat library [3] or in OpenCV [4].
- It is required to add to the comparative study, the effect of extracting features using SIFT versus the other extraction algorithms mentioned above.

12. Rules

- The same groups of lab. tasks (any change in group members is not allowed).
- **All** the group members must be aware of the project (anyone can be asked in any part of the project).
- **IT'S NOT ALLOWED TO USE ANY AVAILABLE SOURCE CODE or LIBRARIES IN IMPLEMENTING CLASSIFICATION ALGORITHMS.**
- You can use any programming language such as **(C++, C#, Java, or Matlab)**

13. Deliverables

- Report that describes the project, system evaluation and comparative study.
- A CD contains the source code, and the report.

14. Deadlines

- Delivering the whole project during the Pattern Recognition practical exam time.
- You must obtain **permission** to use your **segmentation** method by ***Thursday, 7 December 2017***.

15. Support

There will be a support lab for answering your inquiries.

16. References

[1] ImageNet, http://image-net.org/index

[2] David G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, 60, 2, pp. 91-110, 2009.

[3] A. Vedaldi and B. Fulkerson, VLFeat: An Open and Portable Library of Computer Vision Algorithms, 2008, http://www.vlfeat.org/

[4] OpenCV, http://opencv.org/

Good luck! ☺