IJCAT - International Journal of Computing and Technology, Volume 3, Issue 3, March 2016
ISSN : 2348 - 6090
www.IJCAT.org

Object Detection and Recognition in Images

[1] Amruta Chavan , [2] Dipali Bendale , [3] Radha Shimpi , [4] Pradnya Vikhar

[1,2,3,4] KCES's COEIT, North Maharashtra University,
Jalgaon, Maharashtra, India

Abstract - Object Detection and recognition is an important task in image processing and computer vision. It is concerned with determining the identity of an object being observed in an image .Humans can recognize any object in the real world easily without any efforts. But computerize recognition of object in image is not easy task. In such system some problems are occurred such as lightning, mirroring, rotating. It also used Sketch based system for object detection in which user can draw the images by hand and then matches the images from the database. The system developed has many types of applications in the field of Medical Diagnose, Cartography, and Robotics.

Keywords **- Multi-object Detection, Object Recognition, Object Recognition Applications.**

1. Introduction

The Object Detection and Recognition system In Images is web based application which mainly aims to detect the multiple objects from various types of images. It also recognizes the images after performing the detection. In this chapter, The Background and applications of project discussed. Object detection is a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos. Well-researched domains of object detection include face detection and pedestrian detection. Object detection has applications in many areas of computer vision, including image retrieval and video surveillance. Object recognition is an important task in image processing and computer vision. It is concerned with determining the identity of an object being observed in an image from a set of known tags. Humans can recognize any object in the real world easily without any efforts; on contrary machines by itself cannot recognize objects.

The propose is to develop sketch-based method for image retrieval in which users draw sketches via a Web browser that enables the automatic retrieval of similar images from a database of images. The characteristics of these images are different from those of naturalistic images. There are unique challenges associated with such content based retrieval. The propose is to develop sketch-based method for image retrieval in which users draw sketches via a Web browser that enables the automatic retrieval of similar images from a database of images. The characteristics of these images are different from those of naturalistic images. There are unique challenges associated with such content based retrieval. In this, fig 1 shows object detection and fig 2 shows sketch detection.

Fig. 1 Object Detection

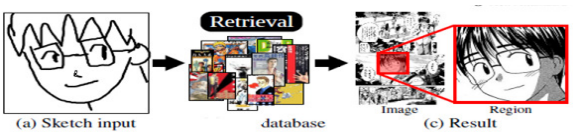

Fig. 2 Sketch Detection

2. Background

The goal of object detection is to detect all instances of objects from a known class, such as people, cars or faces in an image. Typically only a small number of instances of the object are present in the image, but there is a very large number of possible locations and scales at which they can occur and that need to somehow be explored. Each detection is reported with some form of pose information. This could be as simple as the location of the object, a location and scale, or the extent of the object dined in terms of a bounding box. In other situations the pose information is more detailed and contains the parameters of a linear or non-linear transformation. For example a face detector may compute the locations of the eyes, nose and mouth, in addition to the bounding box of the face. Object detection systems construct a model for an object class from a set of training examples. In the case of axed rigid object only one example may be needed, but more generally multiple training examples are necessary to capture certain aspects of class variability.

2.1 Limitations

Lightning: The lightning conditions may differ during the course of the day. Also the weather conditions may affect the lighting in an image. In-door and outdoor images for same object can have varying lightning condition. Shadows in the image can affect the image light. Whatever the lightning may be the system must be able to recognize the object in any of the image.

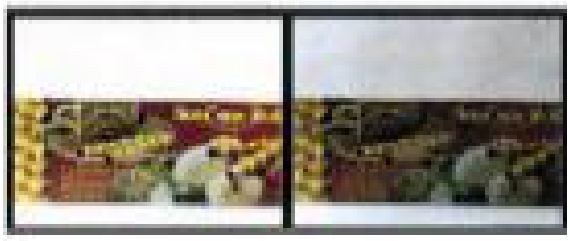

Fig. 3 Difficulties in Lightning

Positioning: Position in the image of the object can be changed. If template matching is used, the system must handle such images uniformly.

Rotation: The image can be in rotated form. The system must be capable to handle such difficulty. The character „A" can appear in any of the form. But the orientation of the letter or image must not affect the recognition of character „A" or any image of object

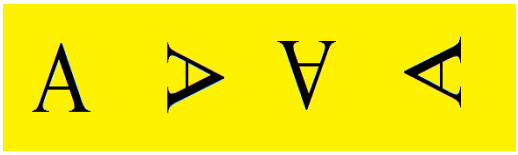

Fig . 4 Different Rotation

Mirroring: The mirrored image of any object must be recognized by the object recognition system.

Occlusion: The condition when object in an image is not completely visible is referred as occlusion. The image of car shown in a box in fig.3.3 is not completely visible. The system of object recognition must handle such type of condition and in the output result it must be recognized as a car. [1]

Fig. 5 Occultation

2.2 Scope of Object Detection and Recognition

The project has wide scope in multiple areas and can easily increase its utilization by adding more efficient algorithms. Some of the areas are as follows-

Medical Diagnose: Use of object detection and recognition in medical diagnose to detect the X-Ray report, brain tumors.

Shapes recognition: Recognize the shape from whole region in images.

Cartography: The cartography as the discipline dealing with the conception, production dissemination and study of maps.

Robotics: In robotics use of object detection is movement of body parts and motion sensing.

3. Methodology

3.1 Architecture Of Object Detection

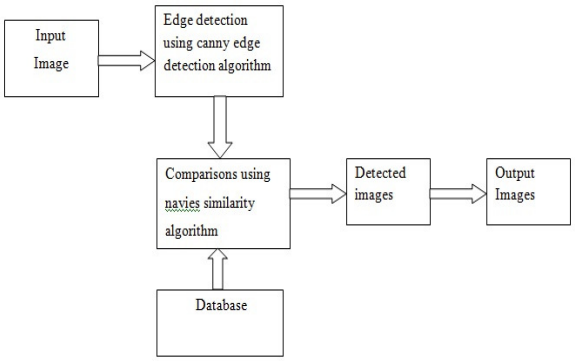

Fig. 6 Architecture Of Object Detection

Input Image:-User takes the input as image.

Edge Detection: - Edge detection is a process of finding the number of edges from the given input image. Canny edge detection algorithm is used to find edge of image.

Navies Similarity Algorithm:- This algorithm is used for matching the images from database/.

Database: - Database helps to detecting the images.
Detected Images:-The system gives the number of similar images as a output after comparison.

Output Image :User Get the Detected images.

4. Implementation

We are develop our proposed system by using java technology and also used the imagevary database to

show the result of our system. It is a standard database. This Database helps to detect the multiple images. This system supports all versions of windows operating system. We implement our system by using two algorithm.1) Canny Edge Algorithm 2) Navies Similarity Algorithm. Canny edge algorithm is used to find the edges of object. Navies' similarity algorithm is used for comparison between database images and user input image.

4.1 Canny Edge Detection

The purpose of edge detection in general is to significantly reduce the amount of data in an Image, while preserving the structural properties to be used for further image processing. Several algorithms exists, and this worksheet focuses on a particular one developed by John F. Canny(JFC) in 1986 .Even though it is quite old, it has become one of the standard edge detection methods and it is still used in research.

The aim of JFC was to develop an algorithm that is optimal with regards to the following criteria:

Detection: The probability of detecting real edge points should be maximized while the probability of falsely detecting non-edge points should be minimized. This corresponds to maximizing the signal-to-noise ratio.

Localization: The detected edges should be as close as possible to the real edges.

Number of responses: One real edge should not result in more than one detected edge one can argue that this is implicitly included in the first requirement.

The algorithm runs in 5 separate steps:

1. **Smoothing:** Blurring of the image to remove noise.
2. **Finding gradients:** The edges should be marked where the gradients of the image has large magnitudes.
3. **Non-maximum suppression:** Only local maxima should be marked as edges.[3]
4. **Double thresholding:** Potential edges are determined by thresholding.
5. **Edge tracking by hysteresis**: Final edges are determined by suppressing all edges that are not connected to a very certain (strong) edge.[3]

4.2 Navies Similarity Algorithm

The features for our test will be 25 RGB triples, corresponding to the average of the RGB values on the 25 regions marked in the figure on the left. The image will be normalized to 300x300 pixels. No texture or variance feature will be stored, only the colour averages. Each region has 30x30 pixels. Each image will be represented, then, a 25x3 feature vector. To calculate the similarity measure between two images A and B we will take each of the 25 regions, calculate the Euclidean distance between the regions and accumulate. The distance from A to A will be, by definition, zero. The upper bound (maximum possible distance between two images, using this similarity measure method) is calculated as 25*(Math.sqrt ((255-0)*(255-0)+ (255-0)*(255-0)+ (255-0)*(255-0))) or a little bit over 11041. This method was chosen because it is simple to understand and implement and can be easily modified by the reader. It combines color (spectral) information with spatial (position/distribution) information, and is expected to be more robust (i.e. tolerant to differences) than comparing pixel by pixel or the average of the whole image, but, again, it is very simple and cannot be expected for perform well in any circumstances, being shown only as an example. To test the feature extraction and similarity measure we will use a set of 24 test images, shown below. Some of those images have similar objects on them (wall, trees, sky) but we are not considering meaning on the images, just patches of similar colors. Images are in different sizes, click on the icons to get the full images. The 16 images on the first two rows are photos of objects, while the last row is of images from the first two rows distorted in scale, color, position, etc[4]

5. Results

Object detection and recognition in images is a web application. There are three options first is the image search by image, second is the image search by sketch and third is the image is detected by edges. In fig 6, we find the edges using canny algorithm of input image. Fig 7 shows the detected images as output using navies' similarity algorithm. In fig 8and Fig 9 shows the sketch detection.

5.1 Screen Shot

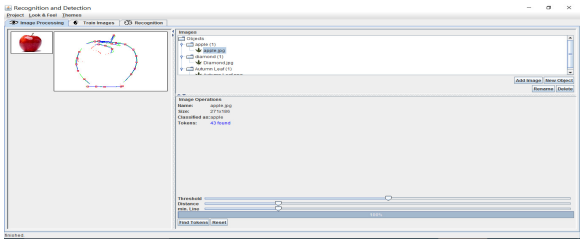

Fig 6 Edge Detected Image

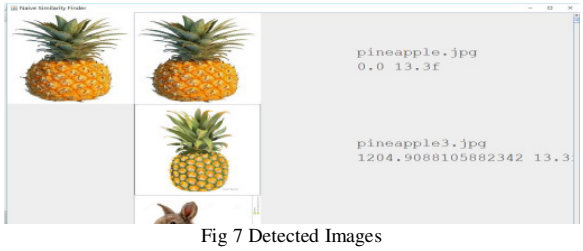

Fig 7 Detected Images

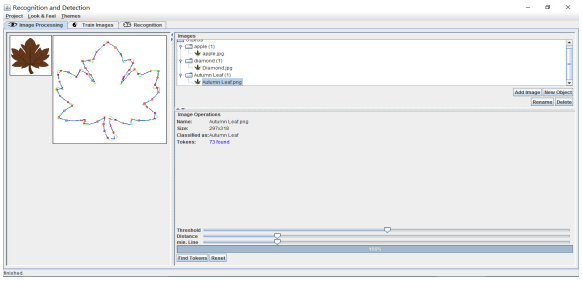

Fig 8 Sketch Edge Detected

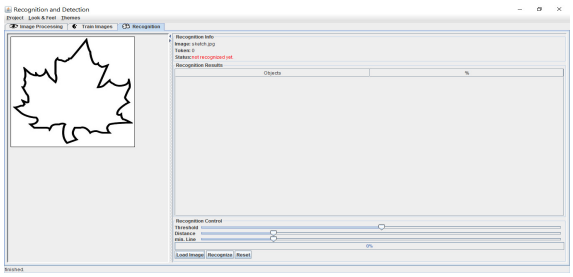

Fig 9 Sketch Detected Images

6. Conclusion

The Object Detection and Recognition system in Images is web based application which mainly aims to detect the multiple objects from various types of images. To achieve this goal shape and edge feature from image is extracted. It uses large image database for correct object detection and recognition. This system will provide easy user interface to retrieve the desired images. The system have additional feature such as Sketch based detection. In Sketch detection user can draw the sketch by hand as an input. Finally the system results output images by searching those images that user want.

7. Future Scope

Object is detected but also tries to find out its location. In Future we will achieve accuracy in motion analysis the segmented moving object from tracking can be further analyzed with the statistics of each motion to verify whether a car is speeding or not, or whether a person is running, walking, or jumping. Processing time need to produce searching time by searching only in some parts of the image. Here use motion trigger and search only in the moving region. Searching algorithm such as hierarchical search or block matching algorithm might be able to make this program faster because it reduces number of pixels to be searched.

References

[1] Reetu Awasthi, "Techniques for Object Recognition in Images and Multi-Object Detection". International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Volume 2, Issue 4, April 2013.

[2] Dilip k. Prasad," Survey of the Problem of Object Detection in Real Images", International Journal of Image Processing (IJIP), Volume (6): Issue (6), 2012

[3] Syeed Irfanullah, "Canny Edge Detection", March 23,2009

[4] Byung hui jeiong, " Navies Similarity algorithm" , ijseia,vol (9),2015