International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-2, July 2019

Recognition and Location Estimation for Multiple Indoor Static Objects

R. Balamurugan, R. Arunkumar, G. Prabakaran

Abstract: ***Locating objects in an image is a very useful task for robotic navigation and visually impaired persons. The ultimate goal of my work is to position the recognized objects in the image. Objects are detected using Adaboost techniques and also recognized from the real-time images. Objects are detected using AdaBoost classifier. SIFT features are extracted from the objects found in the image and classified using Support Vector Machine, and the position of an objects are estimated. We proposed IOLE algorithm to estimate the location of object in an image.***

Index Terms: Object detection, AdaBoost Classifier, Support Vector Machine (SVM) classifier, Scale Invariant Feature Transform (SIFT). Image Object Location Estimation Algorithm (IOLE)

I. INTRODUCTION

Numbers of different approaches have been proposed to detect and recognize the object [1,2]. Object detection and recognition plays an imperative task in robotic applications. The appearance of an object in an image can have a large range of variation due to changes in viewpoint and shape (e.g., non-rigid objects), scene clutter and photometric effects. Different views of the same object can give rise to widely different images.

A. Outline of Work

Locating the Objects in an image consists of; Pre-processing is elaborated in session II. Object detection is described in session III. Proposed Object recognition model is illustrated in session IV and the Object Positioning is explained in session V. Experimental results are described in section VI. Performance measure is depicted in section VII with the conclusion in section VIII.

II. PRE PROCESSING

A. Fuzzy Filter

Fuzzy filter [12], enhance image by filtering both impulse noise and additive noise.

Revised Manuscript Received on 30 July 2019.

***** Correspondence Author

R. Balamurugan*, Research Scholar, Department of Computer Science and Engineering, Annamalai University.

Dr. R. Arunkumar, Associate Professor, Department of Computer Science and Engineering, AnnamalaiUniversity.

Dr. G. Prabakaran, Associate Professor, Department of Computer Science and Engineering, Annamalai University.

B. Gray Scale Conversions

Gray Scale Conversion converts the input colour image into gray level scale images and it is assisted to build the object detection and recognition task easy and convenient.

C. Image Resize

Input images are resized into 2048 by 1586 dimensional, in order to decrease the time complexity and the memory size.

III. OBJECT DETECTION

Finding instances of real-world objects in an image or video is known as object detection [9]. Object Detection Approaches [6] are, Feature-based object, Template-based object, Classifier Based Object, Motion- based Object Detection. Image objects are detected by, extracting features from input image by Haar-Like Features Extraction Technique and then Adaboost classifier are used to depict bounding box around the image objects.

A. Haar - like feature

Haar – like [3], [7], [10] feature can be represented as weighted sum of intensities of rectangular regions in a feature as

$$HF=\sum_{i=1}^{R} sign(i).w_i.\ \mu_i$$

Where,

R - No. of rectangular regions

Sign (i) - sign assigned to the i^{th} rectangular region

w_i - weight (inversely proportional to the rectangular region area

μ_i – Rectangular region average intensity

B. Ada-boost classifier

AdaBoost is an iterative learning algorithm that only utilizes a training set to construct a "strong" classifier and is a "weak" learning algorithm. It converting the weak classifiers into a strong classifier. With the minimum classification mistake, the learning algorithm chooses a "soft" classifier in each iteration. AdaBoost is adaptive in the sense that subsequent classifiers are misclassified by prior classifiers in favor of those sub-windows.

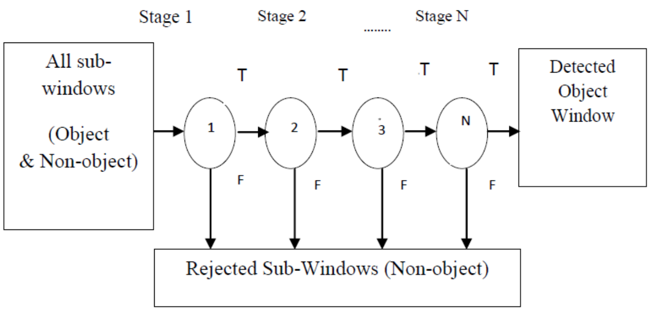


Fig. 2.1 Adaboost classifier

Ada-boost classifier [2,3] (Adaptive Boosting) combines weak classifier algorithm to form strong classifier.

$$A(x)=\text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(X)\right)$$

where,

$h_t(x)$ - Weak classifier output t for input x

α_t - Weight assigned to classifier

$\alpha_t = 0.5 * \ln(1-\varepsilon_t)/\varepsilon_t)$ where ε_t – error rate

Input sequence of n samples

1. Initialize weight of each samples
   $\alpha_{1,i} = 1/n$
2. For x=1 to n do (loop step 3 to 7)
3. Call weak learner h_t with weight α_t
4. Calculate the error ε_t of h_t
   $\varepsilon_t=\sum_{i=0}^{n} \alpha_{t,i} \| h_t(x_i)-y_i \|^2$
5. If error rate $\varepsilon_t > 1/2$, then set T=t-1 goto step
   Set $\alpha_t = 0.5 * \ln(1-\varepsilon_t)/\varepsilon_t)$
6. Update the weights

$$\alpha_{t,i} = \alpha_{t,i} * \begin{cases} e^{-\alpha t}, \text{ if } h_t(x_i) = y_i \\ e^{\alpha t}, \text{ if } h_t(x_i) \neq y_i \end{cases}$$

7. Normalize α_t
   $\alpha_{t+1,i} = \alpha_{t,i} / \sum_{t=1}^{n} \alpha_{t,i}$

Output,

$$A(x)=\text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(X)\right)$$

IV. OBJECT RECOGNITION

Identifying any precise object in a video or an image is the main objective of object recognition [13]. The proposed method recognizes the objects from the scenes through Support Vector Machine. The objects are detected by analyzing the images using Adaboost classifier and segment the region of objects in the scene. This segmented portion is considered as an object image which is undergoing for extraction of SIFT features. SVM model is finally recognize the objects through the extracted features for each object in that scene.

A.. Scale Invariant Feature Transform

Scale-invariant transform function (SIFT) is an algorithm for detecting and recitation local image uniqueness. This algorithm was developed by David Lowe. SIFT can robustly recognize items even between clutter and partial occlusion, as the SIFT function descriptor is not variant to uniform scaling, direction orientation, and also partially in-variant to adjust changes in distortion and lighting. This SIFT descriptor consisted of a technique for identifying points of interest from a gray-level picture where statistics of local gradient directions of picture intensity were collected to provide a summary description of local picture constructions around each point of interest in a local neighborhood. The four stages to find the custom depiction of a object are 1. Detection of scale-space extrema, 2. Key point localization, 3. Orientation assignment and 4. Key point descriptor.

B.. Detection of scale-space extrema

The first step is to build from the input picture a Gaussian "scale space" feature. This is created by converting the initial picture with different width Gaussian functions. The scale space of any image can be defined as a function L(x, y) calculated from the convolution of a variable-scale Gaussian, G(x,y, with an image input, I(x,y):

L(x,y) = G(x,y)I(x,y) (4.1)

Where is the convolution process of x and y, and

$$G(x,y,\sigma)=\frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (4.2)$$

To detect stable main sites in the scale space effectively, Lowe suggested using extreme scale-space in the Gaussian difference of function converted with the picture, D(x, y, σ) which can be calculated from the difference between two neighboring scales separated by a steady multiplicative factor k:

$$DoG(x, y,\sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y)$$

$$= L(x, y, k\sigma) - L(x, y, \sigma) \quad (4.3)$$

C. Key point Localization

This phase tries to remove some points from the list of main points of the candidate by discovering those that have little contrast and are badly located on an edge. The value of the key points in the DoG pyramid in the extrema is set by:

$$D(z) = D + \frac{1}{2}\frac{\partial D^{-1}}{\partial x} Z \quad (4.4)$$

If the function value at **z** is below a threshold value this point is excluded. A 2x2 Hessian matrix, H, calculated at the location and scale of the key point is helped to locate the curvature. Using this formula, the main curvature ratio can be effectively verified.

$$H=\begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (4.5)$$

D. Orientation Assignment

A 16x16 square is chosen in this implementation. The orientation histogram of 36 bins covers the 360 degree range of orientations. Then gradient magnitude, m(x, y), and orientation, θ(x, y) is pre-calculated through pixel differences

$$m(x,y)=\sqrt{(L(x+1,y)-L(x-1,y))^2 + (L(x,y+1)-L(x,y-1))^2} \quad (4.6)$$

$$\theta(x,y) = \tan^{-1}\left(\frac{L(x,y+1)-L(x,y-1)}{L(x+1,y)-L(x-1,y)}\right) \quad (4.7)$$

Each sample is weighted by its magnitude of gradient and a circular window weighted with Gaussian weight. This calculates the place, orientation and scale of the SIFT characteristics observed in the picture.

E. Key point Descriptor

At this stage, a descriptor is calculated for the local image region as distinctive as possible at each applicant main stage. A Gaussian weighting function with π connected with the main point scale is used to allot a weight to the magnitude. The image gradients were joined with an orientation histogram. Each histogram contains 8 arrow instructions and is calculated from 4x4 sub-regions. Each arrow's length refers to the sum of the region's gradient magnitudes close to that direction.

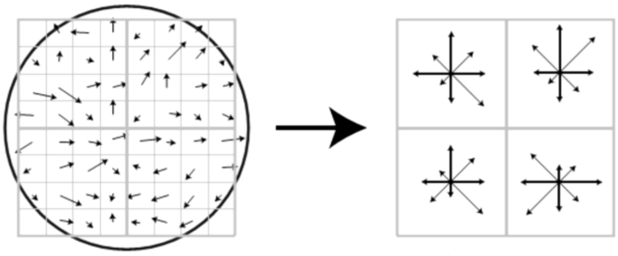


Fig. 4.1 Building key points

F. SVM

Support Vector Machine [7] classifies various types of objects by means of a hyper plane [2]. SVM is a type of supervised learning algorithm, in which classification involves two phases namely Training and Testing. Training phase impart knowledge, about different types of object by means of SIFT features extracted from an image, to the classifier. During the Testing phase, SVM classifier applies knowledge gained from training stage to classify different types object in an image. SVM provides robust classification performance against noise.

$g(x) = \omega^t x + b$

V. LOCATING THE OBJECT

Finding the position of the object will be more usable for the number of applications, especially for robotic applications and visually impaired persons, to move around the object. Location of the object in an image, directs the robot.

A. IOLE

In our proposed work, image objects bounding box dimensions are used to find the location of the objects. Input image area is partitioned into nine sub image areas shown in fig.5.1.

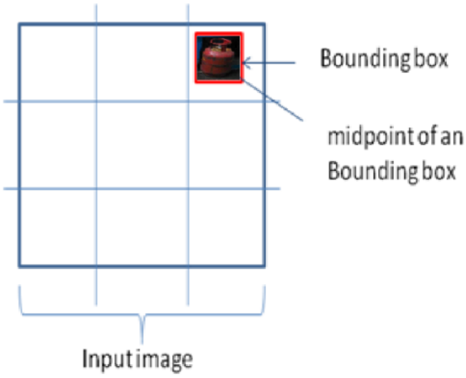


Fig. 5.1 Example of image partition

Midpoint of the bounding box is estimated and compared with image partition coordinates as described in the following algorithm, to estimate the location of object in an image. In this example the object is in the Upper Right corner of an image

B. IOLE Process

1. Input image space is portioned into nine segments, namely 1.Upper left, 2. Upper middle, 3. Upper right, 4. Center left, 5. Center middle, 6. Center right, 7. Lower left, 8. Lower middle and 9. Lower right.
   1.1. Find the width and height of an image
   1.2. Initialize m and n, i.e. the number of horizontal and vertical partition
   1.3. Segment the image area into m * n segments.

2. Estimate the center of the bounding box of a recognized object

3. Compare the value of the i^{th} bounding box center to the input image segments coordinate values to find the exact location of the recognized object bounding box center in the image.

   3.1. If the bounding box center y_i is lesser then than or equal to image first segment height, then do the following,
      i. If the bounding box center x_i is lesser than or equal to image first segment width, then Object is in Upper left Coordinate.
      ii. Otherwise, if the bounding box center x_i is greater than or equal to image Second segment width, then Object is in Upper Right Coordinate.
      iii. Otherwise, if the bounding box center x_i is lesser than or equal to image Third segment width, then Object is in Upper Middle Coordinate.

   3.2. Otherwise If the bounding box center y_i is greater than or equal to image first segment height and lesser than the third segment height then do the following
      i. If the bounding box center x_i is lesser than or equal to image first segment width, then Object is in Center left Coordinate.

   ii. Otherwise, if the bounding box center x_i is greater than or equal to image Second segment width, then Object is in Center Right Coordinate.
   iii. Otherwise, if the bounding box center x_i is lesser than or equal to image Third segment width, then Object is in Center Middle Coordinate.

3.3. Otherwise, if the bounding box center y_i is greater than or equal to image Second segment height then do the following
   i. If the bounding box center x_i is lesser than or equal to image first segment width, then Object is in Lower left Coordinate.
   ii. Otherwise, if the bounding box center x_i is greater than or equal to image Second segment width, then Object is in Lower Right Coordinate.
   iii. Otherwise, if the bounding box center x_i is lesser than or equal to image Third segment width, then Object is in Lower Middle Coordinate.

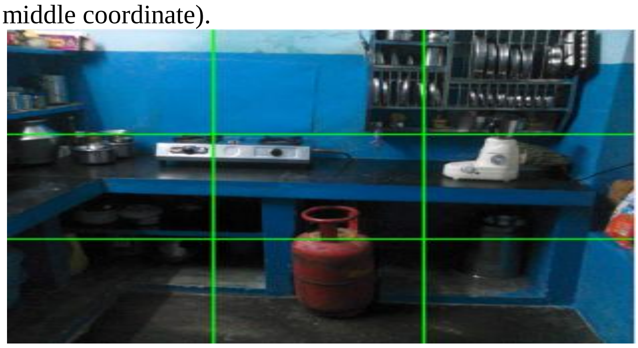
VI. EXPERIMENTAL RESULTS

Real time images with different objects are taken for object detection, object recognition and locating objects in an image.

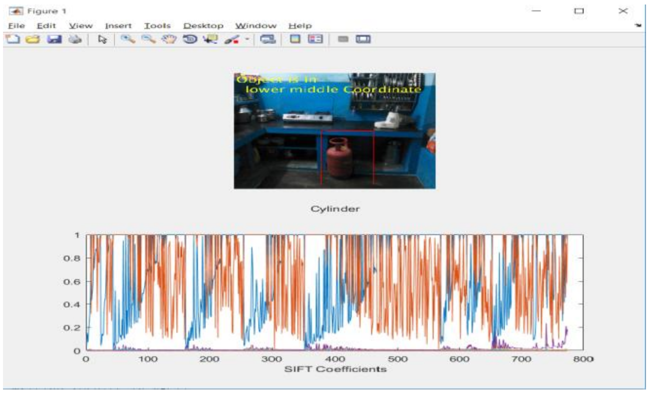
6.1 Original image

Fig. 6.2 shows nine partitioned example image of original image in fig. 6.1, indicated by green line.

Fig. 6.3 shows a recognized object cylinder and its location in an image is displayed as (Object is in lower middle coordinate).

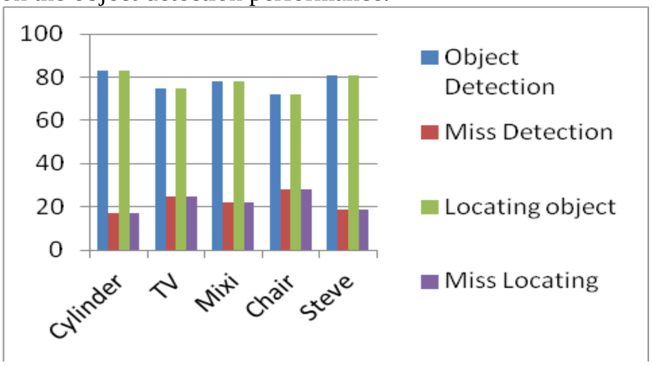


6.2 Example for partitioned mage -object (cylinder) is in lower-middle coordinate of an image.



6.3 Recognized object of an image with location

The performance of this proposed IOLE algorithm is depends on the object detection performance.



6.4 Performance of Object detection and location identification

VII. CONCLUSION

Object location estimation performance is highly depends on the object detection. We can get accurate object locating performance if we got accurate object detection performance, this will help the visually impaired peoples to move around the object after detection and recognition. In future work, we will focus on, developing very high performance object detection and recognition techniques to exactly locate all types of objects and to locate multiple objects in a single image.

REFERENCES

1. Wu, Jian, et al. "A Comparative Study of SIFT and its Variants." Measurement science review 13.3 (2013): 122-131.
2. R. Arunkumar, M. Balasubramanian, G.Vandhana “Detection And Recognition Of Real-Time Indoor Objects Using Sift Features “, Advances in Natural and Applied Sciences, Pages: 269-275 2015.
3. Liu, Yang, et al. "A novel rotation adaptive object detection method based on pair Hough model." Neurocomputing 194 (2016): 246-259.
4. Barinova, Olga, Victor Lempitsky, and Pushmeet Kholi. "On detection of multiple object instances using hough transforms." IEEE Transactions on Pattern Analysis and Machine Intelligence 34.9 (2012): 1773-1784.
5. Liu, Chunsheng, Faliang Chang, and Chengyun Liu. "Cascaded split-level colour Haar-like features for object detection." Electronics Letters 51.25 (2015): 2106-2107.
6. Mubashir, Muhammad, Ling Shao, and Luke Seed. "A survey on fall detection: Principles and approaches." Neurocomputing100 (2013): 144-152.

7. Medjahed, Seyyid Ahmed. "A comparative study of feature extraction methods in images classification." International journal of image, graphics and signal processing 7.3 (2015): 16.
8. Fu, Keren, et al. "Salient object detection via color contrast and color distribution." Asian Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012.
9. Mousavian, Arsalan, Jana Košecká, and Jyh-Ming Lien. "Semantically guided location recognition for outdoors scenes." 2015 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2015.
10. Borji, Ali, and Laurent Itti. "Human vs. computer in scene and object recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014.
11. Lee, Jin Han, et al. "Outdoor place recognition in urban environments using straight lines." 2014 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2014.
12. Frigui, Hichem, et al. "Robust and fuzzy preprocessing algorithms for target detection in ladar range images." 1998 IEEE International Conference on Fuzzy Systems Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98CH36228). Vol. 1. IEEE, 1998.
13. Rani, Ritu, Ravinder Kumar, and Amit Prakash Singh. "A comparative study of object recognition techniques." 2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS). IEEE, 2016.
14. Arunkumar, R., M. Balasubramanian, and S. Palanivel. "Indoor Object Recognition System using Combined DCT-DWT under Supervised Classifier." International Journal of Computer Applications 82.3 (2013): 16-21.

Retrieval Number: B2951078219/19©BEIESP
DOI: 10.35940/ijrte.B2951.078219
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication