International Journal on Science and Technology (IJSAT)

E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

"An Efficient Object Detection Using Faster R-CNN"

Mrs. Surekha Aswale[1], Dr. Raju Sairise[2]

[1,2]Department of Electronics and Telecommunication,
Yadavrao Tasgaonkar College of Engineering and Management, Mumbai University
[1]smadake05@gmail.com, [2]rsairise566@gmail.com

Abstract-

This project presents an efficient object detection algorithm in digital images. Traditional object detection methods, particularly those based on deep learning, offer high accuracy but are often resource-intensive, making them unsuitable for deployment on low-power or real-time embedded systems. To address these chal- lenges, the proposed algorithm focuses on object detection using Otsu threshold- ing, followed by a window-based scanning mechanism and Faster R-CNN boundary expansion technique to accurately identify and isolate object boundaries. The al- gorithm is implemented using Python and Open CV, and tested on real-world datasets, such as traffic images containing closely packed vehicles. It offers a tun- able trade-off between detection resolution and processing time, making it adapt- able for various application needs. The system operates independently of object shape or size, and provides a robust solution for environments where computa- tional resources are limited. Experimental results demonstrate the effectiveness of the approach in achieving accurate detection with minimal overhead. This makes the algorithm well-suited for applications such as traffic surveillance, industrial sorting, and smart agriculture, with future work focused on enhancing robustness to lighting variations and extending support for object tracking in video streams.

Index Terms- Otsu thresholding,Contour detection,Fatser R-CNN,Python,Open CV.

I. INTRODUCTION

Computer vision has brought in many advantages to the society including military and reconnaissance based activities over the past decade. Various object segmentation and detection approaches are used to extract different features of image so as to infer the contents of an image [2]. These detection algorithms are widely used in motion recognition and detection, 3D object identification, video tracking, image mosaicking and panorama stitching.

To Object detection is a computer vision task that involves identifying and localizing objects in images. It has various applications, ranging from autonomous driving and surveillance systems to augmented reality and image recognition. Python, along with the OpenCV (Open Source Computer Vision) library, provides powerful tools for object detection. OpenCV is an open-source computer vision and machine learning software library that offers various algorithms and functions for image processing. The key innovation of this approach lies in its robust handling of detect the objects in cluttered scenes – a scenario where many traditional algorithms struggle. A shape-based method might have difficulty distinguishing between the individual cars, as their shapes may be partially occluded or blended together. However, the proposed algorithm, by focusing on detect the object using a window-based approach, can effectively separate these objects. By combining pixel-wise analysis, dynamic threshold adaptation, and efficient boundary expansion logic, the algorithm offers a compelling trade-off between detection accuracy, speed, and computational complexity.

II. PROBLEM STATEMENT

Object detection and image segmentation are crucial tasks in the field of com- puter vision, as they enable machines to identify and isolate objects within digital images. These tasks are essential for a wide range of applications, from autonomous driving and robotics to medical image analysis and surveillance. While advancements in deep learning have significantly improved the accuracy and efficiency of these tasks, there remains a gap in the performance of object detection algorithms when applied to low-resource environments or real-time systems.

III.OVERVIEW

This section provides a general overview of proposed system.
The technique for object detection is explained in two sections as scanning and filtering algorithm (1) and object
detection algorithm (2). Fig. 1 shows a flow diagram with
colour based image thresholding, scanning and filtering, object
detection, segmentation and saving to data base as different
stages of operation.

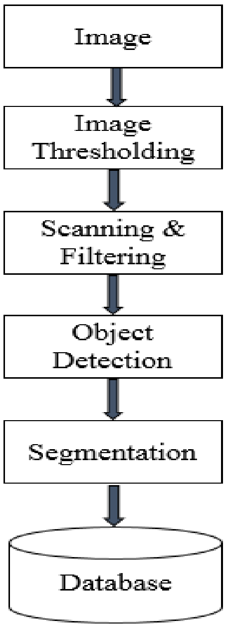

Fig. 1: Flow diagram of proposed algorithm

A. Image Thresholding

Thresholding is a simple yet powerful technique in image processing used to segment or binarize an image by converting it into only two levels — typically black (0) and white (255).

In Otsu's thresholding, the threshold value is automatically calculated by finding the intensity level that minimizes the within-class variance of the image's histogram. This method assumes the image histogram is bimodal, representing two distinct classes of pixels (e.g., foreground and background). The algorithm calculates the optimal threshold (t) that separates these classes, minimizing their combined spread.

B.Scanning and Filtering

The proposed object scanning and filtering algorithm (1) can identify the objects distinctly from each other irrespective of shape and size of the object. Here the object identification does not affect the system accuracy even when the image background is complex. Window based image scanning is chosen to identify the indexes of non-zero values and the windows having more than 50 percent of non-zero values are only considered for object detection.

The binary image is scanned using rectangular windows of a predefined size (h × w). These windows are moved across the image, typically from left to right and top to bottom, with a specified step size. The step size determines the amount of overlap between adjacent windows and affects the speed and accuracy of the detection process.

If the number of non-zero pixels in a window exceeds a certain percentage (e.g., 50%) of the total number of pixels in the window, the window is considered to contain a potential object. This thresholding step helps to filter out background regions that are unlikely to contain objects and reduces the computational effort required for subsequent processing.

Fig. 2: Image scanning with 350x206 pixels window

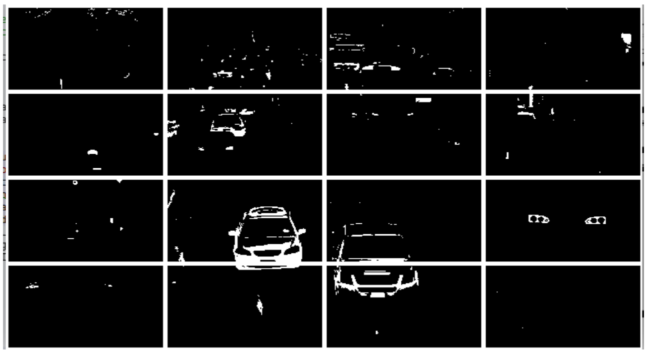

Fig. 3: Image scanning with 150x103 pixels window

C. Object Detection

Object detection is a fundamental task in computer vision that involves identifying and localizing objects within images. This process not only classifies objects but also determines their positions, typically by drawing bounding boxes around them.

The R-CNN (Regions with Convolutional Neural Network features) algorithm is a foundational deep learning method for object detection—a task where both classification and localization (bounding box prediction) are needed.

D. Segmentation

The detected object is segmented from the original image using the bounding box coordinates. The segmented object can be stored as a separate image file or in a database for further analysis.

IV. METHODOLOGY

Object detection utilizes advanced algorithms and neural networks to achieve high accuracy. These methods analyze visual data, extract features, and classify objects, often using techniques such as convolutional neural networks (CNNs) and region-based methods. The proposed methodology involves several key stages, each designed to address specific aspects of the object detection problem.

Without loss of generality let us consider a case where all the red coloured objects in the target image need to be cropped. Using standard thresholding and masking a binary version of the image B is generated where all red coloured objects holds 1's and other pixels assigned with 0's.

A. Object identification algorithm

Notations followed in explaining algorithm.

B – binary image data of H rows and W columns.

Be – B extended by padding zeros of 1 row and 1 column.

p – row index of image pixels ranging from 0 to W.

q – column index of image pixels ranging from 0 to H.

Φhw
pq – window of pixel values ranging from (p,q) to
(p+h,q+w).

Shw
pq – sum of pixel values with in the window Φhw
pq .

p2q2
_
p1q1
B(p, q) – Sum of the pixel values ranging from (p1, q1)
to (p2, q2).

Object identification algorithm (1) sweeps Shw
pq across the
image B systematically and search for non zero values.
Window size [h,w] decides the rate of sweeping and affects the speed of object detection proportionally. On the other hand value of [h,w] is inversely proportional to tightness of boundaries that fits around the object. Window size should be selected such that [h,w] < [H,W] and [H,W] are integer multiples of [h,w] respectively. For each successful detection, pixel indices [p,q] will be passed as argument to object detection algorithm (2).

Data: B - Binary image
Result: pixel indices of top left corner of the detected
objects
Initialization p = q = 1;
while p ≤ H − h do
p = p + h;
while q ≤ W − w do
q = q + w;
if Shw
pq
_= 0 then
Execute object detection algorithm
end
q++;

```
end
p++;
q=1;
end
```

Algorithm 1: Image scanning and filtering

B. Object detection algorithm

Pixel indices [p,q] passed by scanning and filtering algorithm could be considered as top left corner of object detected. Object detection algorithm finds the extent of non-zero values along horizontal and vertical directions of the binary image B to calculate width and height of the object. Once the boundaries of object are detected system proceeds with objection process. This repeats till B is fully scanned.

V. RESULTS AND DISCUSSION

A window image scanning and object detection algorithm was developed and tested on an 206x350 pixel image taken from traffic camera. The Proposed algorithm is developed using Opencv 2.8 library. Relative speed of execution and resolution of object detection for different choices of window sizes can be analysed using table 1. where Tck represents number of CPU clocks pulses.

Fig. 5: Test Image

Fig. 5: Objects detected with 206x350 window size

Fig. 5 and Fig.6 shows various objects detected in the test image Fig. 4, for Different window sizes.

V. CONCLUSION

This paper presents a Faster R-CNN object detection algorithm which detect the objects in an image using otsu thresholding and window scanning, segmentation as preprocessing technique. Relative speed of execution and resolution of object detection for different choices of window sizes shows the increase in number of objects detected with varying CPU clocks. Results shows the flexibility in choosing the resolution of image scanning for object detection against processing time. Work is suitable for the real time target identification and tracking objects in defence and surveillance applications.

REFERENCES

1. N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05), vol. 1, June 2005, pp. 886–893 vol. 1.
2. Anusha Alexander and Meher Madhu Dharmana, “Object detection algorithm for segregating similar coloured objects and database formation,”in 2017 ICCPCT.
3. C. H. Lampert, M. B. Blaschko, and T. Hofmann, “Beyond sliding windows: Object localization by efficient subwindow search,” in 2008 IEEE Conference on Computer Vision and Pattern Recognition, June 2008, pp. 1–8.
4. J. Zhang, K. Huang, Y. Yu, and T. Tan, “Boosted local structured hog-lbp for object localization,” in CVPR 2011, June 2011, pp. 1393–1400.
5. K. van de Sande, T. Gevers, and C. Snoek, “Evaluating color descriptors for object and scene recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 9, pp. 1582–1596, Sept 2010.
6. H. Harzallah, F. Jurie, and C. Schmid, “Combining efficient object localization and image classification,” in 2009 IEEE 12th International Conference on Computer Vision, Sept 2009, pp. 237–244
7. K. E. A. van de Sande, J. R. R. Uijlings, T. Gevers, and A. W. M. Smeulders, “Segmentation as selective search for object recognition,” in 2011 International Conference on Computer Vision, Nov 2011, pp.1879–1886.

8. D. Ioannou and E. T. Dugan, “Parallelogram detection in a digital image with the use of the hough transform,” in Proceedings of 13th International Conference on Pattern Recognition, vol. 2, Aug 1996, pp. 532–536 vol.2.
9. Q. Li, Y. Yu, and R. Hou, “Real-time highway traffic information extraction based on airborne video,” in 2009 12th International IEEE Conference on Intelligent Transportation Systems, Oct 2009, pp. 1–6.
10. P. Viola and M. Jones, “Robust real-time face detection,” in Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001,vol. 2, 2001, pp. 747–747.
11. Z. Wang, Z. Deng, and S. Wang, “Sam: A rethinking of prominent convolutional neural network architectures for visual object recognition,”in 2016 International Joint Conference on Neural Networks (IJCNN),July 2016, pp. 1008–1014.
12. K. Hosoda, M. Watanabe, H. Wersing, E. Krner, H. Tsujino, H. Tamura,and I. Fujita, “A model for learning topographically organized partsbased representations of objects in visual cortex: Topographic nonnegative matrix factorization,” Neural Computation, vol. 21, no. 9, pp. 2605–2633, Sept 2009.
13. A. Mohan, C. Papageorgiou, and T. Poggio, “Example-based object detection in images by components,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 4, pp. 349–361, Apr 2001.
14. A. J. Colmenarez and T. S. Huang, “Pattern detection with informationbased maximum discrimination and error bootstrapping,” in Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No.98EX170), vol. 1, Aug 1998, pp. 222–224 vol.1.
15. M. Tayyab, Z. M. F., and S. S. Ali, “Performance comparison of 2ddiscrete cosine transform and 2d-discrete wavelet transform for neural network-based face detection,” in 2009 International Conference of Soft Computing and Pattern Recognition, Dec 2009, pp. 387–392.
16. B. Moghaddam and A. Pentland, “Probabilistic visual learning for object detection,” in Proceedings of IEEE International Conference on Computer Vision, Jun 1995, pp. 786–793.
17. H. Zuo, H. Fan, E. Blasch, and H. Ling, “Combining convolutional and recurrent neural networks for human skin detection,” IEEE Signal Processing Letters, vol. 24, no. 3, pp. 289–293, March 2017.
18. B. K. Yeo and Y. Lu, “Expeditious diagnosis of linear array failure using support vector machine with low-degree polynomial kernel,” IET Microwaves, Antennas Propagation, vol. 6, no. 13, pp. 1473–1480,October 2012.
19. Y. Guo, J. Wang, W. Zhong, and Y. Gu, “Robust feature based multisensory remote sensing image registration algorithm,” in 2014 Seventh International Symposium on Computational Intelligence and Design, vol. 1, Dec 2014, pp. 319–322.
20. D. Peleshko, Y. Ivanov, N. Kustra, and A. Kovalchuk, “An application of combined detector algorithm to extract the interest points of foreground objects in videostreams,” in 2011 11th International Conference The Experience of Designing and Application of CAD Systems in.