IJARIIT INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 7, Issue 4 - V7I4-1659)
Available online at: https://www.ijariit.com

Surveillance Bot with real time object detection

Aman Agrawal
aman89533@gmail.com
ABES Engineering College, Ghaziabad, Uttar Pradesh

Ajay Suri
ajay.suri@abes.ac.in
ABES Engineering College, Ghaziabad, Uttar Pradesh

Akshay Goel
akshay.17bec1195@abes.ac.in
ABES Engineering College, Ghaziabad, Uttar Pradesh

Aman Varshney
aman.17bec1018@abes.ac.in
ABES Engineering College, Ghaziabad, Uttar Pradesh

ABSTRACT

Computer Vision is a subset of programs and software programs in computer science that can see and understand pictures. Computer Vision has different characteristics, such as image recognition, object detection, processing of images, image processing, etc. Face recognition, car detection, pedestrian counting, online imaging, surveillance systems and self-driving cars are commonly used for object detection. We are able to open a phone, unlock a door just by using a glimpse of our face, and make face recognition work even better than ever before. Computer vision also enables the development of new forms of art. Computer vision is facing increasingly complex challenges and sees more accuracy than people doing similar tasks in visualizing the image. But unlike humans these types of computer vision are incomplete, sometimes making a mistake that can be caused by poor training, low data, incorrect tuning of hyperparameter etc.

Keywords— *YOLO, HAAR, Cascade, OpenCV, COCO*

1. INTRODUCTION

1.1 Simple Convolution Network

It basically works by breaking down images into the smaller groups of pixels called filters. Each filter is made up of matrix or clustering of pixels and the network does a series of calculations like convolution to look for a specific pattern network is looking for, Example: nose, ears, eyes can be extracted using filters which can be used to classify human and animal faces .The training is done with large amount of label training data. Detecting objects is a difficult job as it incorporates the two labelling and training tasks that draw a bounding box around each item of interest in the image and allocate a class label to them. Collectively, all these topics are referred to as identification of objects. Recognition of objects refers to a set of associated tasks in digital images to recognize and detect objects.

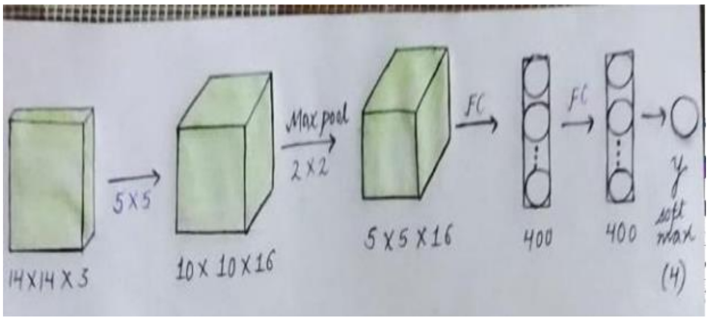
1.1.1 Basic terms used in CNN:

- **Filters**: It is defined as a matrix of pixels which are randomized in beginning. The value of each filter value in matrix is learned during the training process. It is equivalent the weights used in machine and deep learning.
- **Strides:** It is defined as the number of pixels that shifts over the input matrix.
- **Padding:** It is a principle applicable to convolution neural networks and it refers to the amount of pixels that are applied to an image as it is processed for upscaling by the kernel or CNN filter.
- **Pooling:** This method involves moving 2-D filters to each region of the feature map and summarizing the various features within the filter-protected region.

The output dimensions that are just obtained after pooling layer are:

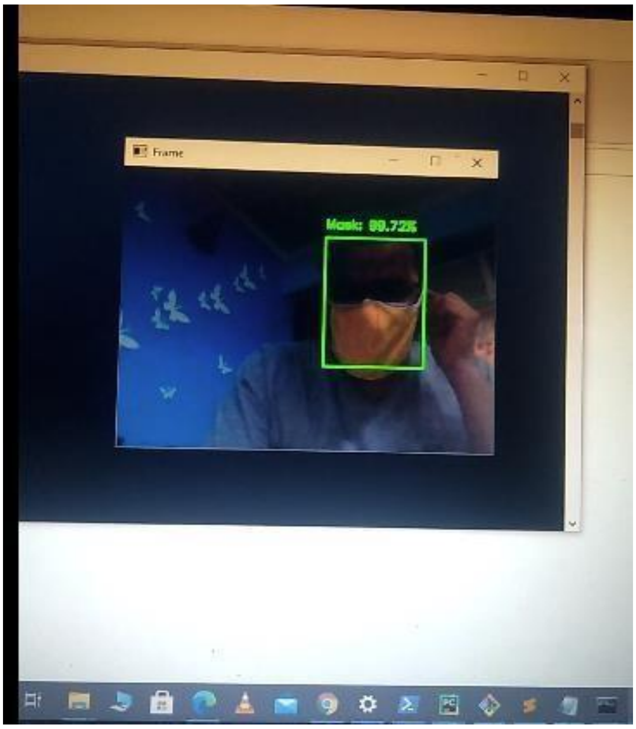
$$(Nh - f + 1) / s * (Nw - f + 1)/s * Nc$$

For a function map with dimensions Nh * Nw * Nc :
Nc : No. of channel
Nh : Height of feature map N
w: Width of features
f **:** Size of filter
s : Stride of filter



1.1.2 Basic steps in CNN:

All the filter values are set when CNN starts, so the first prediction begins to make no sense. Each time a CNN made a prediction based on label data, a function that says how close its prediction was to the right label images was calculated using an error. CNN changes the picture values based on this error or job loss and begins the process again. Each iteration operates with greater precision.



1.2 Object Detection Introduction

Detecting objects is a difficult job as it incorporates the two labelling and training tasks that draw a bounding box around each item of interest in the image and allocate a class label to them. Collectively, all these topics are referred to as identification of objects. Recognition of objects refers to a set of associated tasks in digital images to recognize and detect objects.

Object Localization

Computer vision is one of the fields that is just making remarkable strides and is working so much better than just a few years ago.

The challenges in object detection are location classification, meaning that not only do we have to mark this as the vehicle but algorithm is also responsible for placing a bounding box or drawing a green rectangle in the picture around the position of the car. This thus applies collectively to localization classification. A classification is a system in which an object is categorized based on its pixels. Where the term localization refers to the figuring out of where in the picture is the car we have detected. The term classification basically refers to which class does the object or labelled image belongs to. Example: If we do this for an autonomous driving application, not just other vehicles, but maybe other pedestrians, bicycles, traffic lights, and maybe even other objects might need to be identified.

There is generally one object for the classification and the classification of localization issues. In the middle of the picture, we typically try to identify or recognize and localize one large object. In comparison, several objects can be present in the detection problem. And in fact, within a single image, maybe even multiple objects of different categories. Localization is very helpful to the detection process.

1.3 Landmark Detection:

Facial landmark recognition allows you to detect a number of different points on your face that together make up your eyes, mouth, ears, nose and so on.

From there we're able to apply overlays i.e. filters to get the snap.
There's so much more that it can be used for like emotion analysis and face tracking.

2. LITERATURE REVIEW

2.1 Haar-Cascade Classifier

In an article entitled as, "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001, an effective detection method proposed by Paul Viola and Michael Jones for object detection using Haar Cascade separators. It is an ML-based method in which cascade function is studied from multiple positive and negative images. It is also used in other products to obtain images.

2.1.1 Haar-Features

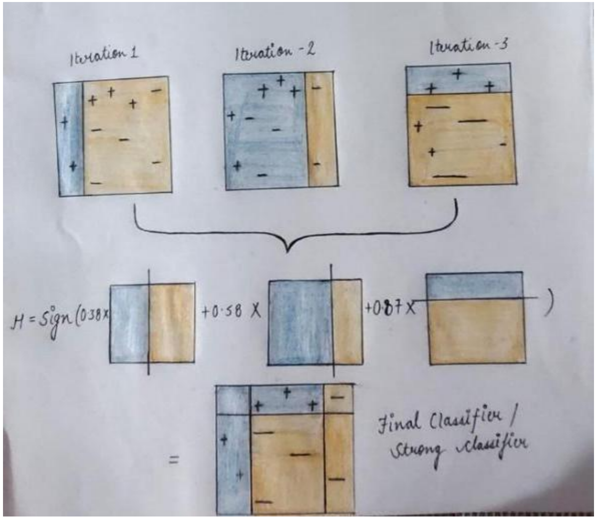
- Haar Features or Haar--Wavelet is a type of sequence of rescaled square shaped functions that is very similar to the fourier analysis.
- It was proposed by Alfred haar in year 1909.
- Haar features are relevant features for face detection and recognition.
- It depends on attributes like face,nose,ear,lips etc.
- Haar features contains two parts:
- Black Part => It depicts the dark part of the face.
- White Part => It depicts the bright part of the face.

Haar features are classified in three categories:

- Line Features => It can detect line features quite effectively.
- Four Rectangle Features => A simple rectangle may be defined hair-like feature as a difference in the pixel dimensions of the rectangular areas that can be in any position and scale in the first image.
- Edge Features => It can detect edge features quite effectively.

2.1.2 Steps involved in Classifier

- Now the pixels will be taken and haar-feature selection will be applied on it. Some of the pixels that will be different in these terms or having low confidence level will enter next step of classifier i.e. Integral image.
- As of now the classifier is still far from predicting accurate and desirable result so in order to improve quality and efficiency we move to third step that is adaboost training.
- The integral image is used to make this classifier process super fast and effective as we can see in above picture there are two good reasons:
- In Adaboost training there will be decision terms that is used to split the correctly classified part of the image and wrongly classified part of the less classified images.
- The positive part would be considered as correctly classified and the negative part considered as wrongly classified region less weightage is given to correctly classified region whereas high weightage to wrongly classified part of image and iteration will be performed on the images and finally strong classifier will come as result.
- The last stage is cascade classifier.



All four stages will be combined together and properly trained classifier is ready to go. Now cascade classifier is already trained, means it is pretrained on desirable positive images and negative images where positive image refers to image we want as output for example in our face detection we want our face as output not a scenery.

2.2 YOLO

2.2.1 Introduction

The YOLOv3 state-of-art object detector is designed to achieve high accuracy and efficiency in real time. YOLOv3, is an upgrade compared to its previous version. It uses a neural network which in a single iteration predicts the location of objects and the class score. This is done by treating the acquisition problem as a deferral problem, which also alters the inclusion of images in their opportunities and their corresponding positions.

The error between expected value and the actual value is determined by the loss function of deep learning network training, using the principle of error backpropagation in the neural network and continuously changing the weight of each layer in the network to complete the model's training

2.2.2 Anchor box

In Faster RCNN, the idea of the anchor box is introduced, and k-means in YOLO V3 are used to find size ratio of the anchor box to detect the target. The parameters relative to the anchor, instead of mapping the coordinates of the bounding box directly, box are predicted. There are total nine YOLO V3 anchor boxes based on the COCO dataset (a large-scale dataset for detecting, segmenting, and captioning objects); the nine clusters are: (10 * 13); (16 * 30);
(33 * 23); (30 * 61); (62 * 45);
(59 * 119); (116*90);
(156 * 198); and (373 * 326).
The finer the grid cell is capable of detecting finer objects, the narrower the receptive field, the greater the size, and the more sensitive it is to small objects. The following nine anchors are therefore allocated by YOLO V3 to the prediction performance of 3 different scales. The scale and the distribution of nine anchors of the COCO data set in YOLO V3.

2.2.3 Loss Function

The error between expected value and the actual value is determined by the loss function of deep learning network training, using the principle of error backpropagation in the neural network and continuously changing the weight of each layer in the network to complete the model's training. The loss function, in this process, defines the path in which the model is trained. The loss function of YOLO V3 is the weighted sum of the loss of position (errors between the expected boundary box and the ground truth), the loss of classification and the loss of confidence (the box object), where the attribute of the localization loss (x, y, w,
h) uses MSE (Mean squared error) and cross-entropy is used by the latter.

Loss = Localization error + Class error + Confidence error

Allocating Bounding Box

For every bounding enclosure, tx,ty,tw and th, the network then it predicts 4 coordinates. If the cell is offset by (cx, cy) from the top left corner of the image and previous bounding box has pw, ph as width & height, so the predictions lead to:

$$bx = \sigma(tx) + cx \quad by = \sigma(ty) + cy \quad bw=pw*e^\wedge tw \quad bh = ph*e^\wedge th$$

We use the squared error loss total during preparation. If the ground truth is t * for any coordinate projection, our gradient is the value of ground truth i.e. (calculated from the box of ground truth) minus our prediction : t* − t* . You will easily evaluate this value of ground truth by reversing the above equations.

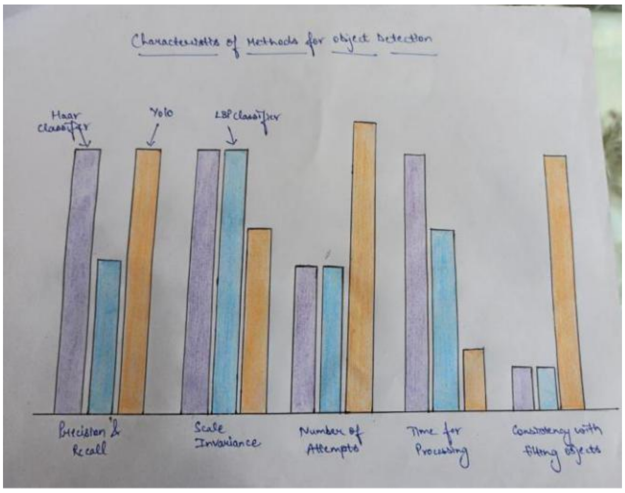
3. METHODOLOGY

There are several ways for artifacts to be observed. Convolutional neural networks, CNN, YOLO, OpenCV, etc. are the best way to detect object methods. In this project, for object detection, we use YOLO and OpenCV methods that clearly detect each and every object. Getting the boundary boxes and label images is the last step. To detect the object, It is easy to grasp and takes less effort. It shows how the objects are identified from the table below and tests how the mechanism goes along to detect an object. To get a proper object image observed, there are four steps to take.
Below are the steps used in this procedure.

Algorithm for proposed method:

Step I : Consider a picture and we need to create a grid that will give us an object's necessary characteristics.
Step II : In this stage, we will use OpenCV to read input image and data points and define the path of the file to an image in the Numpy list.
Step III : Detecting an image in a map view during the OpenCV and Numpy image reading process and transforming the grid to a rectangular box.
Step IV : The final step consists of presenting the picture and the caption on the window with the rectangular box. This is done using YOLO and COCO dataset.

4. COMPARISON AND PERFORMANCE



Comparison Table

YOLO(CNN CLASSIFIER)	HAAR- CASCADE CLASSIFIER
If we manipulate the face i.e., cover eyes or face and also if some features are obsolete YOLO is able to detect it effectively as compared to OpenCV classifiers like HAAR cascade.	Since the division of HAAR Cascade has to be determined manually on the type of accessory, if we provide a separation line and features on the edges where it will only be able to find items with a clear linear and linear properties.
It is like a high level of freedom as it is determined by training and is able to detect blurring or lightly covered faces or objects that rely solely on training and boundaries.	We do not need to train the HAAR features of the cascade subdivision with a small database. The only thing we have to do is train the weight of each element that ends up training all the separations consuming the lowest number of parameters.
Though well trained CNN models like YOLO could detect larger variety of faces but Execution time is slower as Number of parameters are very large.	Haar cascade runs faster and has higher execution speed due to less amount of computations and parameters. Though it lacks in quality performance.
There is quite a hassle in building and training a CNN detection model (YOLO) but the quality of output is outstanding. Pre trained models of YOLO are widely available that can we used easily after a slight tuning.	It is easy to implement due to powerful OpenCv python package which support almost all type of operation like resolution change, live video input and training without writing each function implicitly.
For high performance and minimum training time GPU and high clock frequency processors are mandatory.	It can run on moderate system requirements.

Performance Table

Parameters	**Haar-Cascade**	**Yolo**
Accuracy	Poor	Good
Performance	Poor	Good
Computation Time	Good	Poor
Computation Complexity	Good	Poor

Accuracy Table

Algorithm	**KNN**	**SVM**	**Random Forest**
YOLO v3	0.81	0.78	0.84
Algorithm	**KNN**	**SVM**	**Random Forest**
Haar	0.66	0.64	0.68

5. CONCLUSION

The object can be correctly and accurately detected by the exact position of the object in the picture using the size of x and y, depending on the test results. This paper presents an experimental outcome for two separate object detection methods, namely yolo and hair-cascade, and compares each on the basis of precision, reliability, and efficiency. Both are powerful algorithms that are powerful and are used for modern image acquisition and solid recognition tools as building blocks.

6. REFERENCES

[1] Mohapatra, Hitesh. (2015). HCR (English) using neural network. International journal of advance research and innovative ideas in education, 1(4), 379385.

[2] Mohapatra, H., & Rath, A. K. (2019). Detection and avoidance of water loss through municipality taps in India by using smart taps and ICT. IET Wireless sensor systems, 9(6), 447-457.

[3] Mohapatra, H., & Rath, A. K. (2019). Fault tolerance in WSN through PE-LEACH protocol. IET wireless sensor systems, 9(6), 358- 365.

[4] Mohapatra, H., Debnath, S., & Rath, A. K. (2019). Energy management in wireless sensor network through EB-LEACH (No. 1192). Easy Chair.

[5] Nirgude, V., Mahapatra, H., & Shivarkar, S. (2017). Face recognition system using principal component analysis & linear discriminant analysis method simultaneously with 3d morphable model and neural network BPNN method. Global journal of advanced engineering technologies

[6] and sciences, 4(1), 1-6.

[7] Panda, M., Pradhan, P., Mohapatra, H., & Barpanda, N. K. (2019). Fault tolerant routing in heterogeneous environment. International journal of scientific & technology research, 8(8), 1009- 1013.

[8] Mohapatra, H., & Rath, A. K. (2019). Fault- tolerant mechanism for wireless sensor network. IET wireless sensor systems, 10(1), 23-30.

[9] Swain, D., Ramkrishna, G., Mahapatra, H., Patr, P., & Dhandrao, P. M. (2013). A novel sorting technique to sort elements in ascending order. International journal of engineering and advanced technology, 3(1), 212-126.

[10] Broumi, S., Dey, A., Talea, M., Bakali, A., Smarandache, F., Nagarajan, D., ... & Kumar,

[11] R. (2019). Shortest path problem using Bellman algorithm under neutrosophic environment. Complex & intelligent systems, 5(4), 409-416.

[12] Viola, P., & ones, M. (2001, December). Rapid object detection using a boosted cascade of simple features. Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001 (Vol. 1, pp. I-I). IEEE.