Library Progress International
Vol.44 No.2, July-December 2024: P.508-524

Print version ISSN 0970 1052
Online version ISSN 2320 317X

Original Article

Available online at www.bpasjournals.com

Optimizing Object Detection with Bi-dimensional Empirical Mode Decomposition (BEMD) based Dimensionality Reduction and AlexNet

Dr. J. Anvar Shathik.[1]& Dr. Krishna prasad K[2]

[1], Post Doctorate Fellow, Srinivas University, Mangalore, Karnataka, India

[2]Professor, Dept.of CSE, Institute of Engineering &Technology, Srinivas University, Karnataka,

anvarshathik@gmail.com, krishnaprasadkcci@srinivasuniversity.edu.in

How to cite this article: J. Anvar Shathik., Krishna prasad K (2024) Optimizing Object Detection with Bi-dimensional Empirical Mode Decomposition (BEMD) based Dimensionality Reduction and AlexNet. *Library Progress International*, 44(2), 508-524.

Abstract

Object detection is among the most significant and widely used for identifying target items in a specific image and determining their position and category in order to understand computer vision. Google, Facebook, and Snapchat's server-side production systems have greater freedom to optimize for accuracy, but they are still constrained by throughput limits. Lot of approaches is proposed in the literature for obtaining solution for solving object detection issue. Best results have been obtained by using deep learning and computer vision based methods. However, most of the existing techniques perform under expectation particularly in detecting the small and dense objects and in the detection of objects which has random transformations in geometric measures. The sparse representation methods are often failed to perform particularly in cases where the rate of recognition is more. The problem of object detection can be represented in three stages namely the representation, dimension reduction and object detection. The primary goal of this work is to serve as a guide for the selection of a detection architecture that achieves the proper speed, less memory and accuracy balance for a chosen application. Initially, a dictionary is used for the representation of samples undergoing test followed by sparse representation. Secondly, in order to resolve the issue of high dimension of the features, Bi-Dimensional Empirical Mode Decomposition (BEMD) is introduced for dimension reduction. Thirdly, an architecture called AlexNET is proposed for getting the multi-scaled feature and to add convolutional structure for the detection of dictionaries for the identification of objects which is used as a guideline for the selection of architecture for achieving performance measures such as speed, memory and accuracy. Visualization of images from the Common Object in Context (COCO) dataset offers a side-by-side comparison with current approaches that trace the accuracy/speed tradeoff. The proposed classifier is efficiently reducing number of computational cost and number of parameters when compared to other methods.

Key words: Object Detectors, Bi-dimensional Empirical Mode Decomposition (BEMD), Deep learning, AlexNet, Sparse Reduction, and Common Objects in Context (COCO) dataset.

1. INTRODUCTION

Object detection is a primary research problem in computer vision (CV), and there are several approaches for dealing with the problems in the literature. Object detection is critical for identifying one or more targets in an image or video. It encompasses a wide range of techniques like image processing, pattern recognition, and machine learning (ML). These have a broad range of applications, including traffic control and accident prevention [1], factory alerts, military-related

monitoring, and advanced human-system interaction [2, 3]. Because the application area is typically used in multi-target settings, balancing the relationship between accuracy and cost has become challenging.

The process of object detection is conventionally established through manual extraction of the models of feature, where in the commonly identified features are indicated by Histogram of oriented Gradient(HOG) , SIFT an Haar – based features and other classical techniques that rely on grayscale. SVM based methods or Adaboost techniques are normally used for the classification for obtaining target data. The traditional methods have the ability to determine only the low-level characteristics like contour and texture information. It does not have the ability to determine the multi levels of targets having complex scenarios due to very poor performance.

Numerous researches have been made in the recent past on the issues of object detection owing to the usage of CNNs. These are ML models that come under the Deep learning. In order to perform image recognition, the CNNs can make use of the image information as inputs without pre-processing and additional extractions of features. Hence Conventional Neural Networks (CNN) is very much suitable for the extraction of features in images which include shapes, textures and topology. Advanced methods like Fast R-CNN [4] R-FCN [5],SSD [6], multi-box [7] and YOLOs [8] are presently good for deployment in consumer based products which normally runs in mobile devices

Server based production techniques which are used in web applications such as Google or Facebook have more importance to accuracy but still subjected to throughput. Advanced methods such as COCO challenge [9] are more optimized in terms of accuracy, often relies on models that ensembles multi-crops based techniques which are very slow to be used practically. Only few researches [5, 7, 8] address the issue of running time in a detailed manner. Another discovery is that while these systems are efficient at achieving frame rate, the complete picture of speed and accuracy was not given, which are reliant on other aspects such as the extractor, input size and image. etc. The researchers have a difficult time deciding on architecture and standard measurements of correctness for the application. When it comes to real-time computer vision, running time and memory consumption are just as important as the mean average precision (mAP). Finding tiny and dense things is challenging, and finding objects that undergo frequent geometric transformations considerably more difficult, with present approaches. Existing approaches simply consider how long it takes to train a model, not how well it performs in the real world.

With an aim of high-dimensional issue, The Bi-Dimensional Empirical Mode of Decomposition (BEMD) is introduced for reducing the dimensions. It is used for serving as a guideline for identifying a suitable architecture which achieves more speed and less memory usage with high accuracy in a specific application. Finally, AlexNet is introduced for the purpose of object detection and the architecture is proposed for obtaining multi-scaled feature and which adds deformable structures. Experiments are carryout to trace the speed vs. accuracy curve for various detection systems which varies in meta-architecture, feature extraction and image resolution, etc.

2. LITERATURE REVIEW

Bell et al [10] introduced Inside-Outside Net (ION) for detecting objects that exploit the information both in and out of the regions of interest. Context information which lays an outside the regions of interest are integrated by the assistance of spatial and RNNs. In the inner side, a skip pooling technique is used for the evaluation of design spaces which provide readers with the overview as which methods are most important. The proposed method improves the most used PASCAL based Visual Object Classes (VOC) method of detecting object from 73.9 to 77.9% mAP. The proposed model has shown an improvement of 19.6% to 33.1% in mAP in challenging Microsoft Common Objects in Context (MS COCO) dataset. The proposed method confirms that usage of ION pays vital role in the multi-scale representation and improves detection of small detection. However, the detection rate is lower for bigger items compared to conventional approaches. Due to this problem, the algorithm's speed will not be met.

Erhan et al [11] proposed DCNNsthat obtains performance of state of the art in numerous image recognitions including ImageNet Visual Challenge of Recognition (ILSVRC-2012). The proposed model is based on the localization which uses a single bound box and a score for the confidence for every object in the image. Such models capture contexts of whole image around the required objects but could not handle the issue of multiple instances in the same object. The suggested techniques utilizes a saliency based NN for object detection that predicts the set of bounding boxes termed as class-agnostics with scores for every box and the corresponding likelihood are calculated . The proposed method inherits a number for the variable of instances and for each class, a generalization is allowed in high level order of the network. The proposed method has the advantage of having a grid in regular grid which predicts the boxes that are written as predictors on the images which has shared parameters. During the network, it is very difficult to extract the location and the class label data in a single feed-forward pass

Fu et al [12] proposed Single Shot Multibox (SSD) method for the introduction of additional contexts in the modern methods for detecting objects. In order to achieve the same, the authors have combined a classifier with a rapid detection framework called as SSD. The proposed SSD with de-convolution layer is introduced with additional contexts for the detection of objects and to improve accuracy, especially for the small objects which calls the result system – DSSD for de-convolution single shot based detector. The scenario of using these contributions were easily describes in the high-level, a traditional method does not give success. The experimental results show that how a potential way forward is achieved in terms of the detection parameters of PASCAL, VOC and COCO.

Kim et al[13] investigated on how accuracy can be achieved in maximum level in case of multi-category object detection in minimizing the cost. The proposed method adapts and combines the recent methodologies. The technique also follows the common pipeline of CNN based feature extraction and RoI classification that compresses with the recent techniques such as truncated Singular Value Decomposition (SVDs). The principle of design adapted here is the "less channeled and more layer" approach with adaption of building blocks which includes concatenated ReLU and Hypernet. The proposed framework is deep as well as thin which are trained with the aid of batch based normalization and with residual enabled connections where the rate of scheduling are based on plateau detection.

Lin et al [14] proposed the technique that uses multi-scale and paramedical hierarchy in deep convolution network for constructing features' pyramid that has marginal additional cost. The top-down approach with lateral biased connections is developed to construct high level semantics based maps in all scales. The proposed Feature Pyramid Network (FPN) proved significantly better when compared with generic methods of feature extraction. The technique uses a fundamental R-CNN model that obtains state-of-the-artoutcomes on COCO models without whistles which includes those from the 2016 challenge winners. Finally, the technique which can execute 5 frames/sec in a single Graphics Processing Unit (GPU) is also done which brings out practical and more accurate solutions for object detection. In contrast to the SSD , CNN, and FPN, which often just state with the objective of attaining certain frame-rate, but don't present a whole idea of the speed or accuracy trade-off, which completely relies on numerous other parameters such as which feature extractor is used, input image sizes, and more.

Shrivatsav et al [15] introduced a simple but powerful approach known as the Online Hard Example Mining(OHEM) for training of ConvNet detection. The authors are motivated with the dataset which were having more count of easy objects and small count of hard objects. OHEM uses a simple algorithm which eliminates the heuristics and extra parameters in a common use. The proposed method consistently increases the performance on standards such as PASCAL VOC 2007 and 2012. Efficiency of proposed technique increases as the size of data increases and yields better results in the MS COCO dataset. Moreover, these are combined with the advances of OHEM which leads in better results with 78.9% mAP. Speed or accuracy tradeoff in object detection was addressed.

Huang et al [16] made a research on different types of trade accuracy for ensuring speed and the use of memory in state-of-art methods of CNN bases object detection system. Range of successful techniques is introduced in the current past but comparisons such as apples-apples were difficult. Unified mode of implementation on the Region Based Convolutional Neural Networks (R-CNN), Region-based Fully Convolutional Network (R-FCN), and SSD methods are adopted by making use of alternate feature extraction and other parameters such as the size of the image within all of the meta-architecture. In one end of the spectra, where the speed as well as the memory are vital, a common detector is introduced which can achieve all the speeds required on real time in a mobile device. As far as the accuracy is concerned, a detector is introduced which can achieve more performance based on the COCO dataset. Large amount of training data and more time to fine tune the parameters is highly required for image recognition in CNN Model.

Wei at al [17] introduced image recognition approach based on DNN with Adaptive and weight based Joint Sparse Representation (D-AJSR). The proposed model is data light weight framework that can categorize and identify objects well within the few training sample. In proposed method, CNN is made utilized for extraction of deep feature on the training samples as well as the testing samples. Then, the AD-WJSR method is adopted for sparse representation where the Eigen vectors are again re-constructed through the calculation of each of the Eigen vector. Aiming at the problem of high dimension, The Principal Component Analysis (PCA) is implemented for dimension reduction. Finally, these models are integrated with sparse model, as well as the common and private features of these images that are extracted from the samples in order to form a dictionary. Object detection is done with sparse based classifiers. The experimental findings reveal that the proposed approach outperforms the existing methods. In some circumstances, the D-AJSR system cannot achieve the requirement of a high recognition rate.

Lu et al [18] proposed an object detection approach based on an improved R-CNN method. The enhanced approach can more rapidly and automatically identify the regions where the things belong and are spread in various complex appliances. The feature enhanced framework named the Deeper Region Proposal Network (D-RPN) is an enhancement model for effective extraction of feature in an object in kitchen appliance scene. The U-shape network is then re-structured with the aid of series of enhancement modules of the feature. The experiments performed best in object detection and have attained a mean average precision of 89%. Experiments also prove that suggested technique achieves more detection accuracy than other modern approaches of object detection. The proposed method also works well when applied to recognition of texts.

Cao et al [19] made analysis on the mainstream based object detection methods and multi scaled deformable object detection framework are suggested for dealing with the issues faced by the present methods. The research analysis demonstrates that the high performance is on par or even better than the state of the methods. Deep methods are introduced for obtaining multi-level features and to add deformable Convolutional structures for overcoming the geometric transformations. Fusions of multi-scaled and multi-level features are generated using sampling for the implementation of final recognition of objects and regress. The experiments prove that the suggested frameworks can improve the efficiency in detecting the small objects that has geometrical deformation which shows a consistent improvement in the trade-offs between the speed and accuracy. In this review it concludes that only few methods were addressing about how to improve the speed and accuracy trade-off but it addressing how to reduce the dimensionality of the features from the dataset in the recent works.

3. PROPOSED METHODOLOGY

In the initial stage, a dictionary is made for representing the samples taken for test and then the sparsed representation co-efficient are used. Next, in order to resolve the issue of high-dimension, the Bi-Dimensional Empirical Mode Decomposition (BEMD) is introduced. Thirdly the architecture named Alexnet is implemented for obtaining the features that are multi-scaled in nature and to which the deformable CNN are added for detecting the dictionary for object identification. Primarily, the four recent Meta-architecture the SSD, R-CNN, R-FCN and the AlexNet are proposed in object detection.

3.1. Sparse Representation

This framework for object detection is then applied gradually implemented for other classification of images and object recognition. All of the training data must be presented as

$$W = [W_1, \ldots . W_k] \in R^{m*n} \tag{1}$$

The results $y \in R^m$ of the class "i" shall be expressed in a linear way by the sample taken for training and the same is represented as W_i as in [20]:

$$y = \alpha_{i,1} v_{i,1} + \alpha_{i,2} v_{i,2} + \cdots + \alpha_{i,n_i} v_{i,n_i} \tag{2}$$

By not including the noise, equation (2) could be written by equation (3),

$$y = Wx \tag{3}$$

To obtain sparsest x, object detectors has to resolve the subsequent ℓ_1 minimization issue by equation (4),

$$x' = \arg\min \left|\left|x\right|\right|_1 \; s.t \; Wx = y \tag{4}$$

Here, x denote co-efficient vectors, $||x||_1 = \sum_i |x_i|$ is in ℓ_1 norm. $\delta_i(x')$ is used for representing the non-zero elements which are selected in the corresponding vector of i^{th} class; Hence, y is determined for classification in class (i) through the minimum residual through equation (5),

$$\text{class}(i) = \arg\min_i ||y - W\,\delta_i(x')||_2 \tag{5}$$

In the above equation, $\|.\|_2$ is ℓ_2 norm and $i = 1, 2, \ldots, k$. Due to the correlation among distinct images in sparse representations, the related images is used as a single set, with each image being represented in terms of a combination of public and private means of features in a particular sparse. Public attributes are common to all images in a set, whereas

the private features are exclusive to each image and are unique. As a result, the j^{th} image will be described by public features and will be able to determine its own private features, which will be represented through equation (6),

$$y_j = z_c + z_j, j \in \{1, \dots J\} \tag{6}$$

Assuming such that the images are split into K number of classes, as there are J images for training in every training image in every category. If a particular image is represented in terms of single dimensional vector, the image on the i^{th} object can be given in terms denoted as $y_i \in \left[y_{i,1}, y_{i,2}, \dots, y_{i,j}\right]^T$. As far as the sparse representations are considered, the j^{th} image of the i^{th} class shall be represented in equation as follows,

$$y_{i,j} = z_i^c + z_{i,j}^i, j \in \{1, \dots J\} \tag{7}$$

In case of $\Psi \in R^{N \times N}$is orthonormal, which can describe training image, the equation (7) is represented as in equation (8),

$$\theta_{i,j} = \Psi z_i^c + \Psi z_{i,j}^i = \theta_i^c + \theta_{i,j}^i \tag{8}$$

Multiplied by Ψ^T, on both the sides, then equation (8) is reduced to equation (9)

$$\Psi^T \theta_{i,j} = \Psi^T \Psi z_i^c + \Psi^T \Psi z_{i,j}^i = \Psi^T \theta_i^c + \Psi^T \theta_{i,j}^i = z_i^c + z_{i,j}^i \tag{9}$$

Combined with equation (7), $y_{i,j} = \Psi^T \theta_i^c + \Psi^T \theta_{i,j}^c$, as a result, equation (10) can be used to express the image's joint representation ,

$$\begin{bmatrix} y_{i,1} \\ y_{i,2} \\ \vdots \\ \vdots \\ y_{i,j} \end{bmatrix} = \begin{bmatrix} \Psi^T & \Psi^T & 0 & \dots & 0 \\ \Psi^T & 0 & \Psi^T & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ \Psi^T & 0 & 0 & \dots & \Psi^T \end{bmatrix} . \begin{bmatrix} \theta_i^C \\ \theta_{i,1}^i \\ \theta_{i,2}^i \\ \vdots \\ \vdots \\ \theta_{i,j}^i \end{bmatrix} \tag{10}$$

Equation (10) can be simplified as to equation (11),

$$y_i = \tilde{\Psi} W_i \tag{11}$$

$W_i$protects the discriminated information, and the same depiction in sparse shall be acquired through resolving the (ℓ_1) minimization of subsequent equation(12),

$$W_i = \arg\min \left|\left|W_i\right|\right|_1 \text{ s. t. } y_i = \tilde{\Psi} W_i \tag{12}$$

Following the inverse transformation, the public features of entire images in object i and the private features of every image may be acquired in the Ψ field, after acquiring W_i by equation (13-14)

$$z_i^c = \Psi^T \theta_i^c \tag{13}$$

$$z_{i,j}^i = \Psi^T \theta_{i,j}^i \tag{14}$$

All the features that are public forms a joint feature D and the same is represented in equation (15)

$$D = [z_1^c, z_2^c, \dots . z_k^c, z_{1,1}^1, \dots z_{1,J}^1, z_{2,1}^2, \dots z_{2,J}^2, \dots, z_{K,1}^k, \dots, z_{K,J}^k] \tag{15}$$

The notations used in the sparse representation are clearly discussed in table 1.

TABLE 1. NOTATIONS USED IN SPARSE REPRESENTATION

Notations	Description
$X = [x_1, x_2, \ldots, x_n](x_i \in R^m, \text{general } m << n), i = 1, 2, \ldots, k$	Input samples with N training images that belong to K number of classes
$W_i = [v_{i,1}, v_{i,2}, \ldots, v_{i,n}], v_{i,j}$	j^{th} no of sample in the i^{th} class
$,j \in 1, 2, \ldots, n_i, n_i$	number of sample in i^{th} class
$y \in R^m$	Result vector from sparse input
$\alpha_{i,j} \in R$	sparse representation coefficient of y
$x = [0, \ldots, 0, \alpha_{i,1}, \alpha_{i,2}, \ldots, \alpha_{i,n_i}, \ldots, 0]^T \in R^n$	Input samples
ℓ_1, ℓ_2	$\ell_1$normalization , ℓ_2 normalization
$\delta_i(x') = [0, \ldots, 0, x_i, 0, \ldots, 0]$	mapping function
$y' = W\delta_i(x')$	Re-constructed using the sparse coefficient
z_c	features that are classified as public
z_j	features that are private of the j^{th} image
z_i^c	features that are public on the images in the i^{th} number of object
$z_{i,j}^i$	own features that are private on the images in the i^{th} number of object
$\theta_{i,j} \in \Psi y_{i,j}$	$y_{i,j}$onthe basis of transformation and Ψ
$\theta_i^c = \Psi z_i^c$ and $\theta_{i,j}^c = \Psi z_{i,j}^i$	parts in terms of public and private denoted by Ψ, respectively
$y_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,j}]^T$ and $W_i = [\theta_i^c, \theta_{i,1}^i, \theta_{i,2}^i \cdots \theta_{i,j}^i]^T$.	Vectors
$\tilde{\Psi} = [A, B], A = [\Psi^T \Psi^T \cdots \Psi^T]^TandB = \text{diag}(A)$	complete dictionary which contains couple of parts;

3.2. Bi-dimensional Empirical Mode Decomposition (BEMD)

The iteration and shifting processes for Bi-Dimensional Empirical Mode (BEMD) are the same as for EMD. The approach is a time-bound analysis method that works well with non-linear and non-stationary data. Key concept is to identify an image's inherent multi-scale variance. Du et al[21] , intrinsic functions are expressed through the equation(16) as follows

$$X(t) = \sum_{i=1}^{n} IMF_i + R_n \tag{16}$$

Here the X (t) indicates the image taken as input and the R_N is residue which is decomposed to Intrinsic Mode Function (IMFs) and residue. The images are considered as 2D matrix as f(x,y).

Definition of Extrema Point

Given a one dimensional input, it is easy to define the required peaks and the valleys of an image as the extreme. The detection of these extremes in a two-dimensional image is then done by searching for them individually in the eight directions along rows and columns, including diagonals, in a tabulated matrix of pixels. When a particular place is considered a peak or a valley in all directions, it is referred to as a two-dimensional extreme. As the BEMD involve more shifting and iteration, the time consumption is more for identifying the extrema. Initially, the extreme along the columns are identified and then the rows which have the extrema values of the corresponding columns are found. This sequence helps in saving the time. A condition is termed exceptional where there is a flat peak in the image. In this scenario, the middle point is only considered.

Interpolation and Smoothness

The goal of the next stage is to obtain the maximum and minimum surface envelops using smoothness and interpolation after obtaining the extreme points. In the classic EMD technique, triangulation is utilised in cubic spline interpolation. If the point is viewed outside the convex hull, the triangulation on the point fails, resulting in its infinity and perhaps causing the dimension reduction border effect. The function, on the other hand, is a distance-based interpolation technique in which N pixel points are pointed to a full N*N matrix is created. After then, the matrix is used to solve a system of equations. As a result, even when applied to problems of equal size, the function tends to be quieter and slower. To address the aforementioned concerns, the Gridoff tool is employed for the extrapolation of the outside convex hull and the generation of smooth and surface envelopes for the reduction of the image's real input matrix.

Stopping Criteria in the Iteration Process

The obtained value for IMFs that raises the BEMD cannot satisfy the requirements of IMF definition in the majority of cases. For judging convergence, a global solution will be presented. When the convergence requirements are small, additional iterative decomposition will not offer excess information. Methods with a stopping condition can also be used as an alternative. The criteria for SD is denoted by equation (17),

$$\mathrm{SD} = \sum_{\mathrm{i}=0}^{\mathrm{x}} \sum_{\mathrm{j}=0}^{\mathrm{y}} \left[\frac{\left| \mathrm{H}_{\mathrm{i},(\mathrm{j}-1)}(\mathrm{x},\mathrm{y}) - \mathrm{H}_{\mathrm{i},\mathrm{j}}(\mathrm{x},\mathrm{y}) \right|^2}{\mathrm{H}^2_{\mathrm{i},(\mathrm{j}-1)}(\mathrm{x},\mathrm{y})} \right] < r \tag{17}$$

Where SD is a stopping criteria. r is defined as a constant which is pre-defined in the range of 0.2 to 0.3. The details o the decomposition methods (BEMD) are given as,

1. To find all the local minimum and maximum of the function f(x,y) in order to reduce the dimension of a original matrix image
2. To perform the interpolation of the surface that uses the extrema from the step 1 for obtaining the maximum envelope surface $E_{max}(x,y)$with the minimum value of envelope surface $E_{min}(x,y)$. Here the extension of image is necessary for avoiding the invalid interpolations caused through boundary criteria's.
3. To compute mean envelope surface $Avg(x,y)$this is dependent on max and min envelope surface.
4. $H_{ij}$is then acquired through subtraction of average envelope from basic signal(x,y), where H_{ij}indicatesjth iterations of ith procedure of shifting.
5. To confirm whether the criteria for stopping is satisfied. If no, them the $H_{ij}$is defined as f(x, y) and to repeat 1 to 4 steps. If the condition for stop is already satisfied , it is possible to obtain the $IMF_i$by equation (18),

$$\mathrm{IMF_i}(\mathrm{x},\mathrm{y}) = \mathrm{f}(\mathrm{x},\mathrm{y}) - \mathrm{H_{ij}}(\mathrm{x},\mathrm{y}) \tag{18}$$

It is called as complete process of shifting. The steps 1-5 are then repeated till the threshold becomes a constant and which indicates that the BEMD process of shifting is complete.

3.3. Object detection

The anchor methodology is intended to minimize the losses in terms of combined regressions and classification. For every anchor, initial and best matching truth box b is proposed. If there is a match found, it is termed as "positive anchor" and assigned it as (1) in a class label $y_a \in \{1, \dots, K\}$, (2) denotes the vector encoding of the box b corresponding to an anchor 'a' and set to class label as $y_a = 0$. If an anchor is a predict box that has the encoding of $f_{loc}(I;\ a;\ \theta)$ which corresponds to the class$f_{cls}(I;\ a;\ \theta)$, here I denotes image and θ denotes model's parameter. In this case, the loss for 'a' is calculated as the weighted total of the location-based and classification-based losses, and it is expressed by equation (19),

$$\mathcal{L}(\mathrm{a},\mathcal{I};\theta) = \alpha.\,1[\mathrm{a\ is\ positive}].\,\ell_{\mathrm{loc}}\big(\phi(\mathrm{b_a},\mathrm{a}) - \mathrm{f_{loc}}(\mathcal{I};\mathrm{a},\theta)\big) + \beta.\,\ell_{\mathrm{cls}}(\mathrm{y_a},\mathrm{f_{cls}}(\mathcal{I};\mathrm{a},\theta)) \tag{19}$$

Here, α, βdenotes balancing localizations weights as well as the losses due to classification. For training the model, Equation (19) is leveraged over the anchors and also reduced in terms of parameter θ. In case of multibox [11], these "box priors" are created by clustering truth boxes of database. These anchors are then constructed using complex algorithms by tiling a collection of diverse boxes in varying aspect ratios and scales on a regular basis throughout the given image. One benefit of having such an anchor is that these predictors will be expressed as tiled predictions of image with shared

parameters, similar to how traditional sliding window approaches are done. The fast R-CNN [4] and the latest multibox[6] are the one to adopt this technique in the preliminary experiments.

3.3.1. Single Shot Detector (SSD)

SSD is used for creating anchors in the same method and choosing the highest convolutional feature map and a higher resolution feature map at a lower level, subsequently count a sequence of convolutional layers by spatial resolution decaying with a factor of 2 with every additional layer used for prediction. The term SSD is used to refer broadly to architectures with the purpose of makes use of a single feed-forward convolutional network towards directly predict classes and anchor offsets without requiring a second stage per-proposal classification operation[22]. From this definition, the SSD meta-architecture has been discovered in a number of precursors as mentioned in recent work [26]. This strategy can be used in both Multibox and RPN stages of Faster CNN [6] to predict the class-agonistic boxes for predicting the final class labels [12].

3.3.2. Faster R-CNN

In the Faster R-CNN algorithm, object detection is performed in two stages. In the first stage, Region Proposal Network (RPN), images are processed with a feature extractor (e.g., VGG-16), and features at some particular intermediate level (e.g., “conv5”) are utilized to predict class-agnostic box proposals. The loss’ functions are taken in the form as in Eq(19) which uses the anchor grids. In the later stage, these proposals are then used for the cropping of features in the same intermediate maps that are consequently fed into remains of extractor (such as in fc7) for class prediction and class particular refinement of every scheme. By employing the suggestions derived from the RPN of anchors, the loss functions for these stages likewise assume the form of eq (19). It's also worth noting that the crop suggestions are extracted directly from the image and re-run utilising feature extractions that can be replicated in computations. Nevertheless, some computations must be executed just once in each region, causing the run time to be dependent on the number of independent regions recommended by RPN.

3.3.3. Region-based Fully Convolutional Networks (R-FCN)

Faster R-CNN is considered as in terms of magnitude which are quicker than R-CNN , the facts that these are region particular components that has to be used many times on an image. The R-FCN methods suggested by Dai et al[15] are similar to the Faster R-CNN methods, rather than cropping features in the same layer, the crops are taken from the last layers of feature prediction. This kind of approach by pushing the cropping into the last layers is then taken from the features before prediction. Cropping is pushed to the last layer, which reduces the amount of per-region computations required. According to Dai et al[5], the object recognition challenge requires localising representations that account for translation variance and so can offer sensitive mechanisms in place of more typical ROI operations[4] and distinct crop techniques. They show that R-FCN method, which employs the Resnet 101, should be able to attain higher accuracy for faster CNN than the running times.

3.3.4. Alexnet CNN deep learning architecture

In this work, the AlexNET architecture with CNN deep learning [22] is introduced for the detection of object. The network is presented in a deeper way than the regular CNN with 5 convolution layers which is followed by three kind of pooling layers like 3CONV with $3 \times 3 \times 3$ &384 Kernels, 4CONV with $3 \times 3 \times 3$ &384 Kernels, and 5CONV with $3 \times 3 \times 3$ &384 Kernels. Three different sizes are used for experimentation and evaluation. Drop out of 0.5 percent can be applied to the full connected layers for avoiding the issue of over fitting. Figure 1 depicts the suggested architecture.

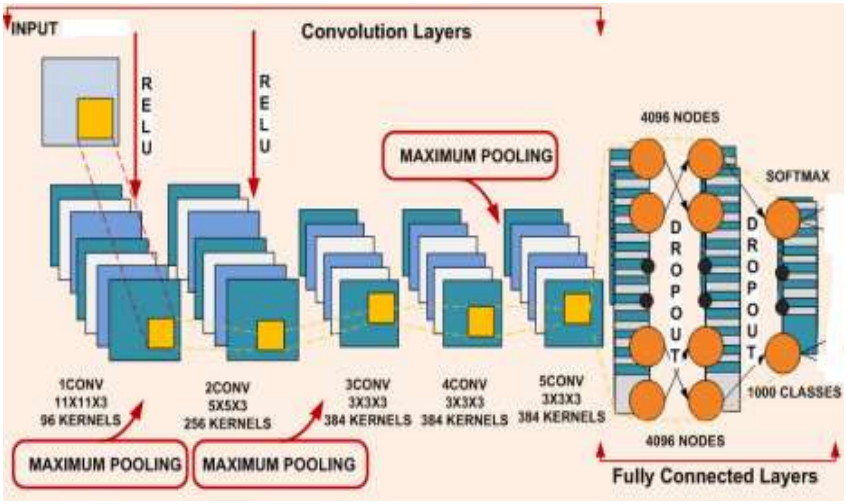


FIGURE 1. THE SUGGESTED FRAMEWORK'S ALEXNET CONVOLUTION NEURAL NETWORK ARCHITECTURE

In this work, the input layer is presented as the pre-processing layer where input images are sampled from the initial size to 227*227 in spatial resolution for reducing the cost of computation. The proposed system then used five CONV layers which are then followed by the three POOL layers and a Rectifying Linear Unit (RELU). A feature map is generated by every convolutional layer. These maps are made up of the first, second, and fifth layers, which are used in various combinations with the 3*3 matrix pooling layers and the 2*2 matrix side. These frameworks are comprised of eight different layered architectures that have 4096 nodes. This generates the trainable feature sets in a feature extraction phenomenon of these layers. These are then subjected to Soft Max activation for the determination of classification probability used for final classification's output. These probabilities which are in the soft max layer can make categories up to 1000 various other classifiers.

3.3.4.1. Convolution Network Layer

This layer is significant in the deep learning context with neural networks which generates feature maps that are confined with the classification. It has a kernel which be able to slide over the image which is taken as input and which generates output called as feature map. In each location, of the input, a matrix multiplication is performed which is followed by the integration of results. The feature map's output is then defined byequation (20),

$$N_x^r = \frac{N_x^{r-1} - L_x^r}{S_x^r} + 1, N_y^r = \frac{N_y^{r-1} - L_y^r}{S_y^r} + 1 \tag{20}$$

Here, (N_x, N_y) denotes the width as well as the height of the expected output of the feature map in final layer, (L_x, L_y) denote kernel size (S_x, S_y) which depicts the total count of the pixels which are skipped through the kernel in an horizontal and vertical direction with index r that indicate corresponding layer , that is, r=1 . The input is then convolutioned, and the kernel is utilized to obtain the outputs, that are specified byequation (21),

$$X_1(m,n) = (J * R)(m,n) \tag{21}$$

Here, $X_1(m,n)$denote a 2D output of the feature map which is got through the convolution of 2D kernel R of the size $(L_x, L_y)$with the feature map denoted by J. The * symbol is used for the representation of convolution between the J& R. The same is represented byequation (22),

$$X_1(m,n) = \sum_{p=-\frac{L_x}{2}}^{p=+\frac{L_x}{2}} \sum_{q=-\frac{L_y}{2}}^{q=+\frac{L_y}{2}} J(m-p, n-q)R(p,q) \tag{22}$$

In the proposed method, there are five layers of CONV along with RELU and the normalization layers are made used for extracting the maximum number of feature maps from the input stage for training the dataset to ensure maximum efficiency.

3.3.4.2. Rectified Linear Unit Layer

In the following stage, the activation of RELU function is done for all the layers which are trainable for strengthening the network for making it as non-linear. It is then applied over the feature map which has been generated from the CL. The use of $tanh(.)$ and RELU function makes the non-linear gradient to get saturated and comes down in terms of training time. The same is expressed by equation (23),

$$X_2(m,n) = \tanh\big(X_1(m,n)\big) = 1 + \frac{1 - e^{-2*X_1(m,n)}}{1 + e^{-2*X_1(m,n)}} \tag{23}$$

where $X_2(m, n)$ is a two-dimensional output feature map following to tanh(.) on the input feature map $X_1(m, n)$, which is attained following passing during the convolutional layer. Values in final feature map are obtained after the completion of RELU activation function; it is represented by equation (24),

$$X(m,n) = \begin{cases} 0, if\ X_2(m,n < 0) \\ X_2(m,n), if\ X_2(m,n \geq 0) \end{cases} \tag{24}$$

Here, the $x(m.n)$ is achieved through the transformation of negative inputs to zero and then return back the same on receipt of any positive values. The RELU layer is then considered to be the proposed framework as the deep neural networks are trained much faster and are intact with the RELU layers.

3.3.4.3. Maximum Pooling Layer

The pooling layer is then included in the architecture proposed after the initial and final convolution and then through the 5th CONV for decreasing the spatial size of every image for bringing down the cost of computation of the framework proposed. The operation for pooling simply picks the value which is maximum for every slice on the image. The application of the pooling layer which is on the activation's output for the purpose of down sampling is shown in figure 2.

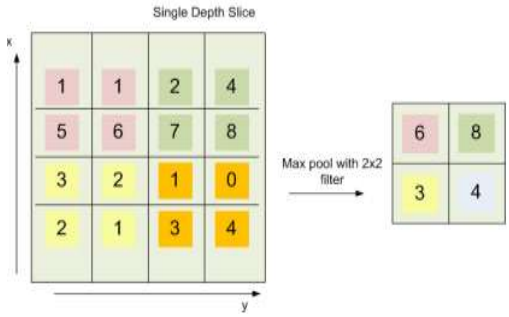


FIGURE 2. MAXIMUM POOLING LAYER

In the proposed method, the pooling is applied for using the maximum values against every slice and the same brings better results with this mechanism.

3.3.4.4. Response Normalization Layer and the Softmax Activation

This is performed after two successful sessions for reducing the error while testing of the proposed networks. The layer helps in normalizing the input layers inside the network all along with the inputs of whole network and the same is described by equation (25),

$$N_{e,f}^{x} = \frac{b_{e,f}^{x}}{\left(z + \alpha \sum_{j=\max\left(0,x-\frac{c}{2}\right)}^{\min\left(T-1,x+\frac{c}{2}\right)} \left(b_{e,f}^{x}\right)^{2}\right)^{Y}} \quad (25)$$

Here, the $N_{e,f}^{x}$represent the activity of normalization consisting of $b_{e,f}^{x}$number of neurons which are computed in position (e, f) within the use of Kernel K. T is the total range within the layers z,c and α are the given constants which are hyper-parameters of adjustable values through application of validation sets respectively. The soft-max is a classifier on the top of the features extracted. After performance of five series of CNNs, the output is then fed to soft-max layer for multi-classification which helps in determining the probabilities of the classification. These are in turn used by the final classification layer for classifying the input images into different classes.

3.3.4.5. Dropout Layer

This layer is then applied to the two layers which are fully in connection when the total count of the iterations gets doubled in the network for avoiding the overfitting of data by substantially increasing the count of iterations through a factor of two that makes the neurons dense. It then performs the average model along the neural network which shows efficient result in regularizing the data which are trained. The maximum number of layers that are used for pooling and the size of the kernel are processed in such a way that the output maps of the feature are down-sampled into single pixel per map. Figure 3 then depict the regularization of methods on layers that are full connected prior and after the drop out being applied.

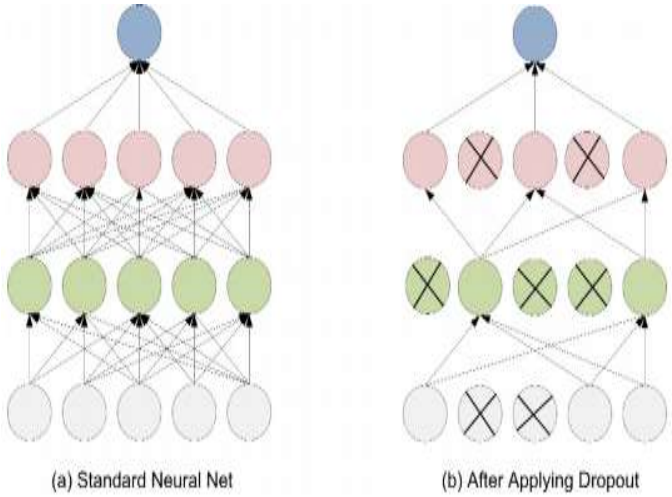


FIGURE 3. FULLY CONNECTED LAYERS (FC) BEFORE AND AFTER APPLYING DROPOUT

The layers that are fully connected also get connected to the output of the top layers in single dimension vector. The upper layers are always fully connected to that of the output for a given class labels such that the high level features are formed from the training data.

3.4. Matching

The targets of classification are determined for every anchor which requires matching anchor for instances of ground truth. Greedy Bi-Partile or many-one matching techniques where the Jaccard method is not employed however the matching's kept ignored if the overlap value is very low are two prevalent ways. The studies employ a throughput with given thresholds that are proposed in the current survey on every meta-architecture, referring to these techniques as bipartite or Argmax. After the matching method is completed, a standard sampling procedure is used to bring the overall count of positive and negative anchors to a specified ratio. Recommended ratios are set for each meta-architecture in the investigations.

4. RESULTS AND DISCUSSION

Here analyses the data which has been collected during the training and benchmarking and to sweeping over the model configurations. Every model includes a selection of meta-architectures with stride feature extractors (for Resnet and Inception), and the input resolution and number of suggestions. These models were trained in some images scaled upto M number of pixels in the shorter edges where as in SSD, these images are resized in a particular shape defined by m*m. The evaluation models are then explored for down scaled images as to trace for the accuracy and speed. In particular, these images are trained through different peaks of resolutions in each model. The experimental setup has M=600 and for high resolution and M is set to 300 for low resolution images. In any scenario, the SSD should only process fewer pixels on average than a faster R-CNN, Alexnet, or R-FCN model with all other variables maintained constant. These methods are implemented via the use of MATLAB environment. The networks are then trained using the COCO data, with entire training images being used in place of image validation. For every model, the time is measured in GPU, the memory required and the precision and recall are calculated. The ones listed below are the most important. True positive (TP) implies that a ground-truth bounding box was correctly detected. False Positive is the instances of incorrect detection of a non-existent object in an existing object. It is represented as FP. False Negative refers to situations in which the ground-truth bounding-box remains undiscovered. It is important to highlight that a TN does not apply in the context of object detection because there are an infinite number of bounding boxes that do not need to be detected in a given image. Assessment of detection mechanism are normally based on the parameters such as precision P, recall R and the same is represented by equation (26,27,28),

$$P = \frac{TP}{TP + FP} = \frac{TP}{Alldetections} \tag{26}$$

$$R = \frac{TP}{TP + FN} = \frac{TP}{Allgroundtruths} \tag{27}$$

$$\text{F-measure} = 2\frac{P*R}{P+R} \tag{28}$$

Precision is considered as the ability of a model for the identification of each relevant object. It is the percentile of the accurate positive count of predictions. Recall is the model's ability for understand and find all the cases that are relevant. It is calculated as the percentile of accurate predictions out of all the given truths. The harmonic mean of recall and precision together is called F-Measure. In all the interpolations, it is seen that the one can interpolate via all the points in such a way that are defined by equation (29,30),

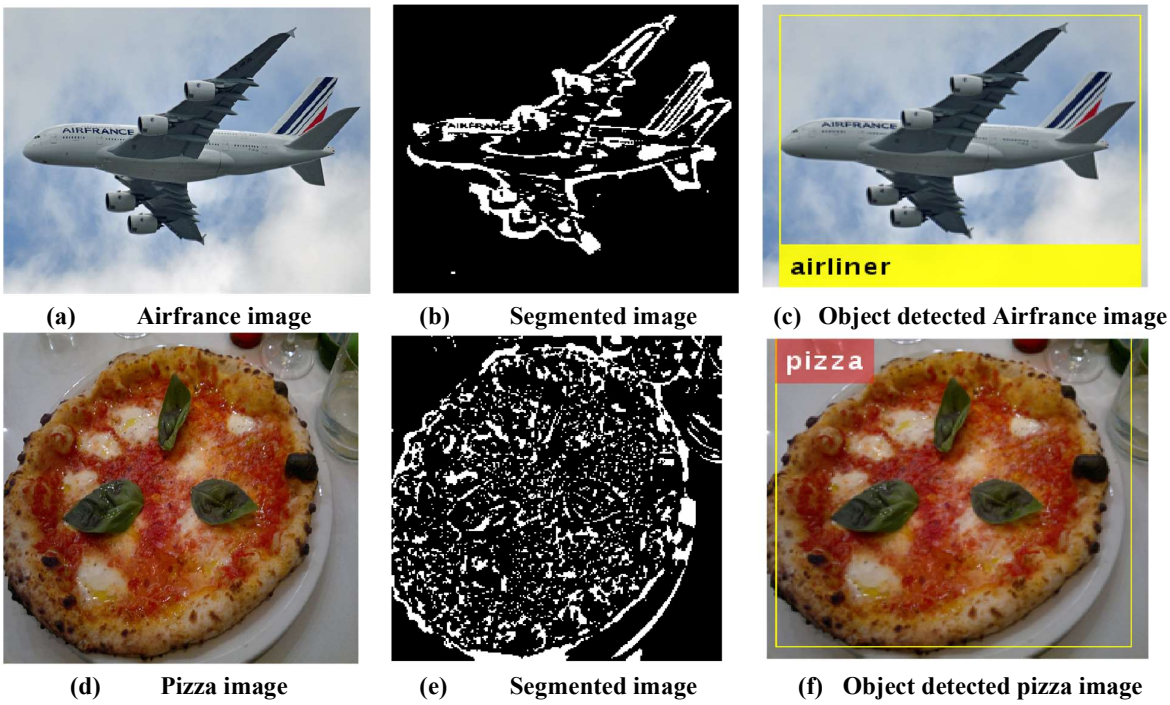
$$AP_{all} = \sum_{n} (R_{n+1} - R_n) P_{interp}(R_{n+1}) \tag{29}$$

$$P_{interp}(R_{n+1}) = \max_{\tilde{R}:\tilde{R} \geq R_{n+1}} P(\tilde{R}) \tag{30}$$

Rather than using precision obtained by a few points, the AP may now be derived by interpolating precision at each level, which takes the highest amount of precision for assessing object detection accuracy in all classes of a dataset. The mAP is the average AP among all classes [4], [7], and is defined by equation (31)

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \tag{31}$$

N represents the number of classes being assessed, and AP_i is the AP in the i^{th} class.In the figure 4(a-c) shows the input image, segmented image, and object detected of airfrance. In the figure 4(d-f) shows the input image, segmented image, and object detected of pizza.



(a) Airfrance image (b) Segmented image (c) Object detected Airfrance image

(d) Pizza image (e) Segmented image (f) Object detected pizza image

FIGURE 4. IMAGE RESULTS

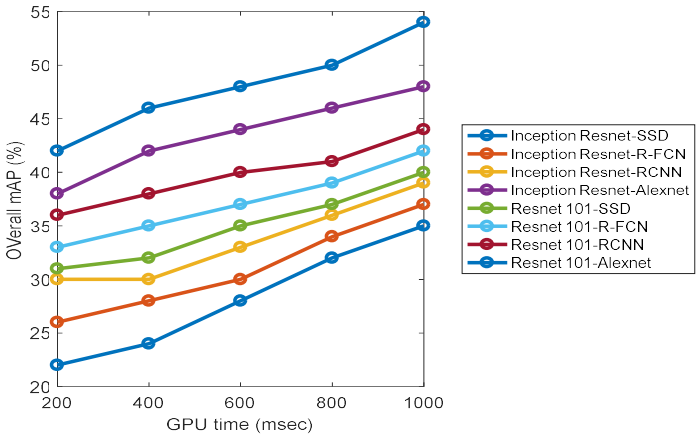


FIGURE 5. ACCURACY VS TIME COMPARISON WITH RESPECT TO FEATURE EXTRACTORS

Figure 5 depicts some visualization of MAP in each of classifiers along with feature extractors. It is seen from findings that it concluded that suggested technique RCNN and AlexNet type of models are fast when compared to the R-FCN. In terms of accuracy R-FCN holds an upper hand. When the GPU time is increased, Alexnet with aid to Resnet, the feature extraction has greater Map as that when compared to the inception of Resnet extraction model. The proposed Resnet 101-Alexnet gives higher mAP value of 54% for GPU time of 1000 milliseconds, whereas other methods such as Inception Resnet-SSD, Inception Resnet-R-FCN, Inception Resnet-RCNN, InceptionResnet-Alexnet, Resnet 101-SSD, Resnet 101-R-FCN, and Resnet 101-RCNN gives mAP value of 35%, 37%,39%, 48%, 40%, 42% and 44% respectively. The proposed classifier is more accurate than the previous approaches because sparse reduction is used in the system, which reduces the system's runtime and boosts the system's accuracy.

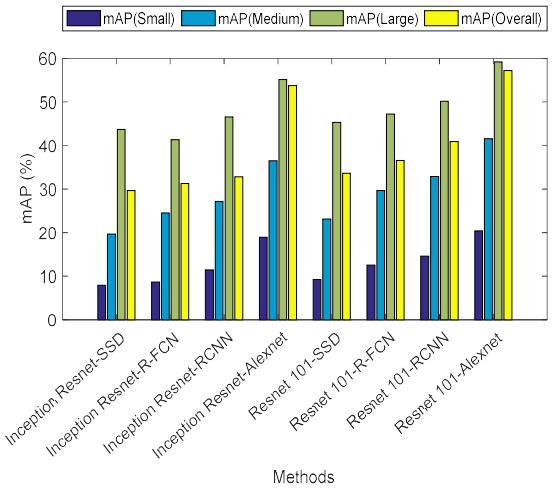


FIGURE 6.ACCURACY STRATIFIED BY OBJECT SIZE, META-ARCHITECTURE AND FEATURE EXTRACTOR, IMAGE RESOLUTION TO 300

In figure 6, the performance analysis is done on the different models with various objects' sizes. It is not surprised to see that entire techniquesoutperforms in case of large objects. It can be inferred that SSD approaches perform badly for small objects, but are comparable with faster RCNN and R-FCN for large objects, outperforming these Meta architectures for faster and lighter extractors. Proposed Resnet 101- Alexnet gives higher mAP value(Large) of 59.21%, whereas other methods such as Inception Resnet-SSD, Inception Resnet-R-FCN, Inception Resnet-RCNN, Inception Resnet-Alexnet,Resnet 101-SSD, Resnet 101-R-FCN, and Resnet 101-RCNN gives mAP value of 43.67%, 41.36%,46.58%, 55.12%, 45.32%, 47.21% and 50.15% respectively. The mAP value of proposed system is higher due to sparse reduced features via Bi-Dimensional Empirical Mode Decomposition (BEMD).

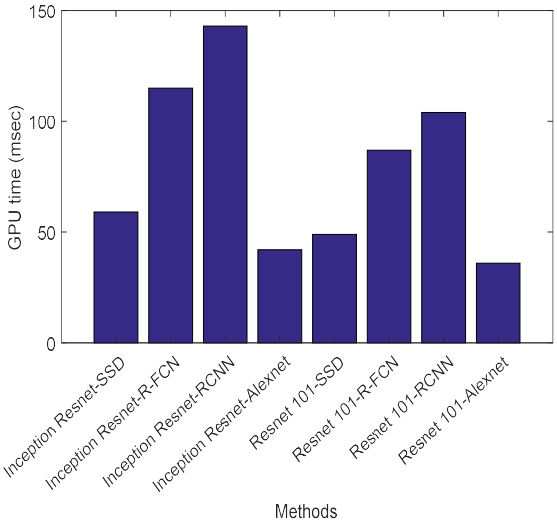


FIGURE 7. GPU TIME (MILLISECONDS) FOR EVERY MODEL, FOR IMAGE RESOLUTION OF 300

In figure 7, a plot of GPU time for each of the model is shown. This has a constraint of platform dependency. Due to a number of issues like caching, I/O, and optimization, the counting and floating point operations offer us with platform independent measurements for the computation, which might or might not be linear in terms of the exact operating cost and time frames. The GPU time of the proposed Alexnet and the inception Resnet has faster GPU time of the order of 42 and 36 Milliseconds. The GPU value of proposed system is faster by reducing the dimensions of the features in the dataset.

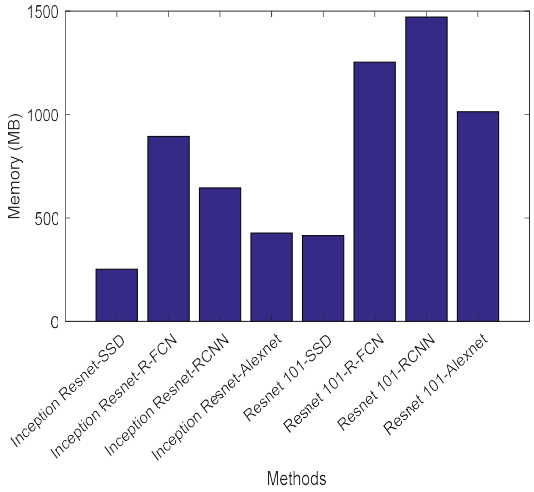


FIGURE 8. MEMORY (MB) USAGE FOR EACH MODEL VS. FEATURE EXTRACTION METHODS

The figure 8 measures the total usage of memory other than the peak memory usage. These error bars reflect the variance in the usage of memory by the various proposals for faster R-CNN. By drilling down the meta-architecture and feature extractor selections, the figure depicts a few of the related informations in further detail. It can be concluded from the figure that the proposed Alexnet with the resent 101 and the incent has fewer memory usage of the order of 1014 MB and 428 MB which has lesser usage when these are compared to the R-FCN and RCNN respectively.

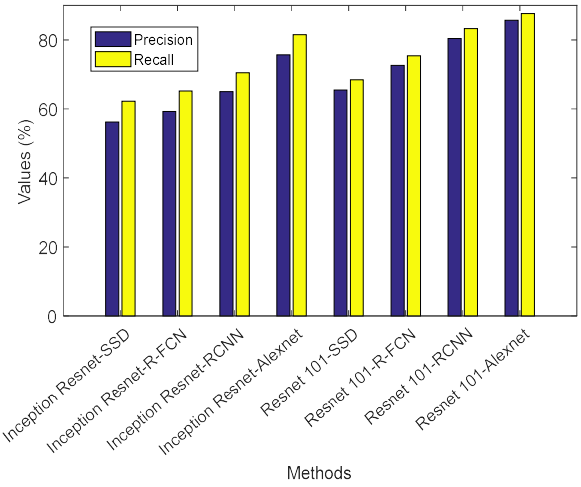


FIGURE 9. PRECISION AND RECALL RESUTLS COMPARISON OF EACH MODEL VS. FEATURE EXTRACTION METHODS

Figure 9indicates recall and precision comparison by using various numbers of proposals. As seen in the figure it concludes that the proposed Alexnet with Resent 101 and Inception Resnet has higher precision value of 85.7143% and 75.63%whereas other methods such as Inception Resnet-SSD, Inception Resnet-R-FCN, Inception Resnet-RCNN, Restnet 101-SSD, Restnet 101-R-FCN, Restnet 101-RCNN has gives precision value of 56.21%, 59.25%, 64.98%, 65.51%, 72.63% and 80.41% respectively. Number of features in the dataset is reduced via the BEMD may reduces the error of the system it might automatically increases the precision and recall values of the system.

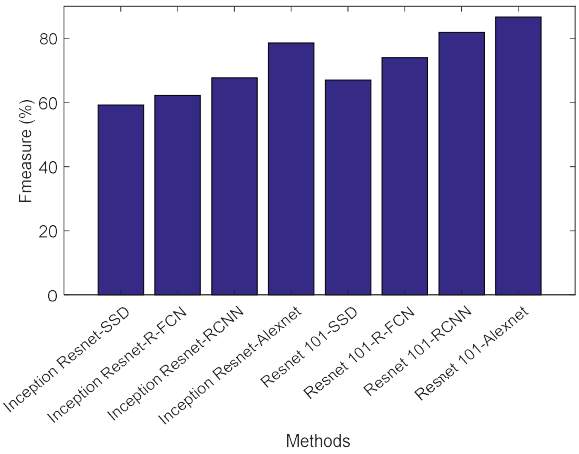


FIGURE 10. PRECISION AND RECALL RESUTLS COMPARISON OF EACH MODEL VS. FEATURE EXTRACTION METHODS

Figure 10 shows the f-measure comparison by using different numbers of proposals. From the figure it concludes that the proposed Alexnet with Resent 101 and Inception Resnet has higher f-measure value of 87.2727% and 78.575% whereas other methods such as Inception Resnet-SSD, Inception Resnet-R-FCN, Inception Resnet-RCNN, Restnet 101-SSD, Restnet 101-R-FCN, and Restnet 101-RCNN has gives f-measure value of 59.23%, 62.235%, 67.745%, 66.995%, 74.02% and 81.825% respectively.

5. CONCLUSION AND FUTURE WORK

Object detection is one of the important and most used methods for the identification of target objects in a specific image and for the determination of the position and category for achieving computer vision's understanding. In this research, the reduction in dimensionality is addressed using the BEMD technique for achieving high order object detection that too in a low space of dimension. The extrema in a two-dimensional image are identified by searching for them separately in eight distinct orientations among various columns, rows, and diagonals as a tabulated pixel. For each object detection, the methodologies which are introduced are to minimize the combined effect of classification and the regression loss. Four Meta architectures, R-FCNs, R-CNNs, and Alexnet classifiers, are used to provide flexible and unified implementations. Alexnet is built in a more complex manner than a typical CNN, with five CNNs and three pooling layers. COCO detection is viewed as a challenge because it gives bounding box coordinates for over 200,000 images representing 80 different object types. This data is made used for the comparison of result with other object detection methods. The performance metrics used are the AveragePrecision (AP), Mean Average Precision (MAP), precision and recall with F-Measure. Results indicate that when fewer proposals are used, the Alexnet can speed up very fast without involving huge losses in terms of accuracy and thus making it comparable with a faster R-FCN, SSD and RFCN.For the detection model to work with complicated input formats, the suggested implementation does not require any changes. As a result, this approach has been used to recognize objects in a variety of complicated input formats. It is expected that the current system will be expanded to handle images of various sizes, and that the quality of the images will be improved. The tuning of the classifier's parameters will play a crucial role in improving object detection outcomes will be left as scope of future work.

REFERENCES

1. Shine, L. and Jiji, C.V., 2020. Automated detection of helmet on motorcyclists from traffic surveillance videos: a comparative analysis using hand-crafted features and CNN. Multimedia Tools and Applications, pp.1-21.

2. Liu, J., Yang, Y., Lv, S., Wang, J. and Chen, H., 2019. Attention-based BiGRU-CNN for Chinese question classification. Journal of Ambient Intelligence and Humanized Computing, pp.1-12.

3. Cao D, Zhu M, Gao L et al (2019) An image caption method based on object detection. Multimed Tools Appl 78(24):35329–35350

4. Ren S., K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91–99, 2015.

5. Dai J., Y. Li, K. He, and J. Sun. R-FCN: Object detection via region-based fully convolutional networks. arXiv preprint arXiv:1605.06409, 2016.

6. Szegedy C., S. Reed, D. Erhan, and D. Anguelov. Scalable, high-quality object detection. arXiv preprint arXiv:1412.1441, 2014.

7. Liu W., D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.- Y. Fu, and A. C. Berg. SSD: Single shot multibox detector. In European Conference on Computer Vision, pages 21–37. Springer, 2016.

8. Redmon, J., Divvala, S., Girshick, R. and Farhadi, A., 2016. You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition , pp. 779-788.

9. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C.L., 2014, Microsoft COCO: Common objects in context. In European conference on computer vision (pp. 740-755). Springer, Cham.

10. Bell S., C. L. Zitnick, K. Bala, and R. Girshick. Insideoutsidenet: Detecting objects in context with skippooling and recurrent neural networks. arXiv preprintarXiv:1512.04143, 2015.

11. Erhan, D., Szegedy, C., Toshev, A. and Anguelov, D., 2014. Scalable object detection using deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition ,pp. 2147-2154.

12. Fu C.-Y., W. Liu, A. Ranga, A. Tyagi, and A. C. Berg.DSSD: Deconvolutional single shot detector. arXiv preprintarXiv:1701.06659, 2017.

13. Kim K.-H., S. Hong, B. Roh, Y. Cheon, and M. Park. Pvanet:Deep but lightweight neural networks for real-time object detection.arXiv preprint arXiv:1608.08021, 2016.

14. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B. and Belongie, S., 2017. Feature pyramid networks for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2117-2125.

15. Shrivastava, A., Gupta, A. and Girshick, R., 2016. Training region-based object detectors with online hard example mining. In Proceedings of the IEEE conference on computer vision and pattern recognition ,pp. 761-769.

16. Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S. and Murphy, K., 2017. Speed/accuracy trade-offs for modern convolutional object detectors. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7310-7311.

17. Wei, W., Can, T., Xin, W., Yanhong, L., Yongle, H. and Ji, L., 2019. Image object recognition via deep feature-based adaptive joint sparse representation. Computational Intelligence and Neuroscience,vol.2019, no. 8258275, pp.1-10.

18. Lu, M. and Chen, L., 2020. Efficient Object Detection Algorithm in Kitchen Appliance Scene Images Based on Deep Learning. Mathematical Problems in Engineering, vol.2020, no. 6641491, pp.1-12.

19. Cao, D., Chen, Z. and Gao, L., 2020. An improved object detection algorithm based on multi-scaled and deformable convolutional neural networks. Human-centric Computing and Information Sciences, 10, pp.1-22.

20. Zhang S., Z. Zhang, W. Kong, and W. Kong, "Combining sparse representation and singular value decomposition for plant recognition," Applied Soft Computing, vol. 67, pp. 164–171, 2018.

21. Du, S., Liu, T., Huang, D. and Li, G., 2018. A fast and adaptive bi-dimensional empirical mode decomposition approach for filtering of workpiece surfaces using high definition metrology. Journal of manufacturing systems, 46, pp.247-263.

22. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems; Neural Information Processing System Foundations- Inc.: USA 2012; pp. 1097–1105.