ADV MATH SCI JOURNAL
Advances in Mathematics: Scientific Journal **9** (2020), no.6, 3759–3775
ISSN: 1857-8365 (printed); 1857-8438 (electronic)
https://doi.org/10.37418/amsj.9.6.54 Spec Issiue on ICAML-2020

COMPARATIVE ANALYSIS OF DEEP LEARNING METHODS FOR OBJECT DETECTION

K. GILL AND V. MANGAT[1]

ABSTRACT. Object detection is a vital field involving machine learning and computer vision. Recent object detectors dependent on deep learning methods are showing assuring results for object detection in images, videos and real-time environment. This paper compares the working of two famous deep learning methods for object detection viz. RCNN and MobileNets SSD. R-CNN uses combination of region proposals and CNNs. Selective search method is used in RCNN technique. Whereas MobileNets SSD is based on depthwise separable convolutions that uses single filter enforced to every input and outputs are combined by using pointwise convolution. A comparative analysis of these two techniques is presented over benchmark and validation datasets.

1. INTRODUCTION

Detecting and tracking objects, [1], is an essential task in the field of computer vision which detects, recognizes and tracks objects in the given set of images. Object detection applications include human interactive games, content-based indexing, security, traffic monitoring and many more. Recent trend in object detection is deep learning which is part of machine learning that includes algorithms inspired by artificial neural network. Convolution neural network

[1]*corresponding author*

2010 *Mathematics Subject Classification.* 68T05, 68Q32.

Key words and phrases. object detection, convolution network, RCNN, MobileNets, MultiBox SSD.

(CNN) is popular in deep learning for object detection. Many techniques exist that are based on CNN as RCNN, Fast RCNN, Mask RCNN. RCNN works on obtaining number of candidate regions, [2], and calculation of convolutional networks, [3], on each region. Advanced work on CNN network includes MobileNets and Single Shot MultiBox Detector (SSD). MobileNets SSD is a technique that uses depthwise separable convolution. This technique is really fast and more accurate than RCNN and YOLO technique, [4] (You Only Look Once). This paper presents a deep learning algorithm combining MobileNets and Single Shot Detector and its comparison with RCNN. Section 2 discusses the prior work done on object detection. Section 3 describes R-CNN technique. Section 4 explains MobileNets, depthwise separable convolutions, network structure and training. Section 5 presents experimental results when MobileNets SSD technique is applied to the images (RGB and thermal) and its comparison with R-CNN technique. Section 6 presents the conclusion.

2. Related work

There are two main types of object detection techniques, viz. sliding windows and region proposal classifiers. Before convolutional neural systems, the main core of these two methodologies: Selective Search, [2], and Deformable Part Model (DPM), [5], were almost identical. After the improvements expedited by R-CNN, [6] region proposal, object recognition techniques became powerful by combining selective search region proposals with convolutional network-based classification. One of the most known pedestrian detector that does not use deep learning features is Integrate Channel Features (ICF) detector, [7] which is based on Viola- Jones framework, [8]. This ICF detector requires boosted classifiers and feature pyramids. Boosting classifiers are building blocks for pedestrian detection. The feature pyramids of ICF have been enhanced in a few ways, that includes ACF, [9], LDCF, [10], SCF, [11], and numerous others. Based on the popularity of R-CNN, [12]; a object detection method that is based on deep learning features; a series of strategies, [11–13], were built for pedestrian detection using two-stage pipeline. SCF detector [11], is utilised to propose regions, trailed by a R-CNN for classification; TA-CNN, [12], utilizes the ACF identifier, [14] to create proposals, and trains a R-CNN-style system to jointly advance pedestrian detection; DeepParts technique, [13], applies LDCF detector, [10],

to produce proposition and takes in an arrangement of corresponding parts by neural systems. These proposers remain stand-alone pedestrian identifiers comprising of hand-made highlights and boosted classifiers. R-CNN approach, [16], is a deep learning technique in object detection to obtain a specific number of candidate object regions and calculate convolutional networks on each region. R-CNN approach has been enhanced in many ways. First methodology enhances the speed and quality of post-classification using SPPnet, [17], by presenting a spatial pyramid pooling layer. Fast R-CNN, [18], expands SPPnet with the goal that it can adjust all layers by limiting loss for confidences and bounding box regression, which was first presented in MultiBox, [19]. Faster R-CNN, [20], extended RCNN with a Region Proposal Network. It has been shown to be flexible and robust, [21].

Second methodology enhances the nature of proposal generation using deep neural systems. In MultiBox [19], [22] the Selective Search proposals are aided by proposition created from a different deep neural system. This enhances the identification precision however it results in complex setup. Faster R-CNN, [20], replaces selective search proposals by ones gained from region proposal network (RPN) and introduces a strategy to coordinate RPN with Fast R-CNN by rotating between finetuning shared convolutional layers. SSD is same as region proposal network (RPN) in Faster R-CNN that utilizes a settled arrangement of (default) boxes for expectation. Rather than utilizing these to pool features and assess another classifier, a score is created for each object classification in each case. In this manner, SSD avoids the complexity of combining RPN with Fast R-CNN and is simpler to implement and faster. Other techniques that are related with SSD avoid the first step and foresee bounding boxes and confidences for various classes specifically. OverFeat, [23], a version of the sliding window strategy, predicts a bounding box specifically from every area of the highest component after knowing the confidences of basic object categories. YOLO, [4], utilizes the entire feature map to predict confidences for numerous classes and bounding boxes. SSD approach is more flexible than the current techniques as it utilizes default boxes of various perspectives. If we utilize only one default box for each area from the feature map, SSD would have related architecture to OverFeat, [23], and if we utilize the entire feature map and include a completely associated layer for predictions rather than our convolutional indicators, we can reproduce YOLO, [24].

MobileNets use depthwise separable convolutions presented in [25] and utilize as a part of Inception models [26] to decrease the calculation in the initial couple of layers. Flattened systems, [27], build a system out of completely factorized convolutions and demonstrate the capability of factorized systems. Another system Squeezenet, [28] utilizes a bottleneck way to plan a small system. An alternate approach for getting small systems is contracting, factorizing or packing pretrained systems. Also different factorizations have been proposed to accelerate pretrained systems [29, 30]. Another strategy for preparing small networks is distillation, [2], which utilizes a bigger system to train a smaller system and it is correlated to our approach. In recent studies an automatic traffic density estimation technique is used, [34], Mobilenet SSD for car counting and performed quantitative analysis between Mobilenet SSD and SSD. Another recent technique is based on convolutional neural network for ground object detection, [35], used for disaster response and recovery.

3. Recurrent convolutional neural network (R-CNN)

R-CNN approach, [16], is a deep learning technique in object detection to obtain a specific number of candidate object regions and calculate convolutional networks on each region. The aim of R-CNN is to take in an image as input and recognise the objects in the image by using bounding box with labels for each object as shown in figure 1. To find these bounding boxes R-CNN uses a number of boxes in the image and check whether and any of them actually matches an object. As R-CNN can be challenging to a particular region proposal technique, selective search is used to allow a limited comparison with previous detection as in [2], [31]. R-CNN uses a Selective Search process to find out bounding boxes (also called region proposals) by using sliding window with CNN. Selective search [2] approach analyses the image by using windows of various sizes and for each window it attempts to gather neighbour pixels by color, intensity or texture in order to identify the objects. After creating the proposals R-CNN encloses the region to a fixed unit size and moves it from a revised version of AlexNet. Feature extraction is done by using 4096-dimensional feature vector, [32].

At the last layer training and testing is done using SVM (Support Vector Machine) with negative mining that identifies the object and its type i.e. to classify

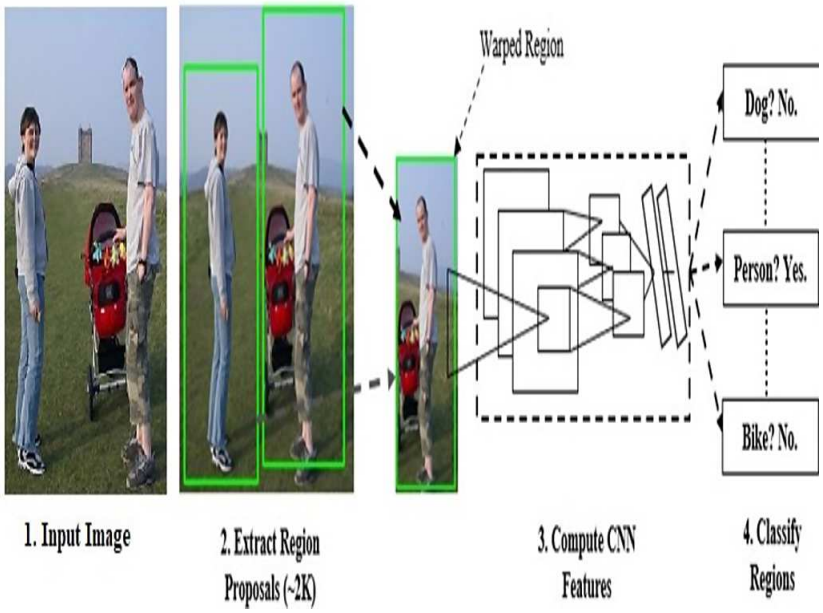


FIGURE 1. R-CNN: Regions with CNN features

objects. It uses shared CNN parameters with low dimensional features.
For the final result, the objects in the bounding boxes the box can be made more secure to fit the real dimensions of object and this can be done by using linear regression on bounding box. For this regression model a set of proposals are generated for bounding boxes, after that it passes the images in the region proposals from a pre-trained AlexNet. Then the SVM classifies the type of object and at last linear regression model is applied.

Figure 2 depicts that object detection by R-CNN approach has three modules:

- First module creates region proposals, [33], that defines the set for candidate detection.
- Second module is based on feature extraction using large convolutional neural network.
- Third module has class of linear support vector machines.

4. MOBILENETS SSD MULTIBOX DETECTOR MULTIBOX

In SSD, the bounding box regression technique is based on MultiBox, [22]. MultiBox as shown in figure 3, is a technique used for quick class-agnostic

bounding box coordinate proposals. MultiBox uses an Inception-style convolutional network in which 1x1 convolutions helps in dimensionality reduction keeping the height and width same.

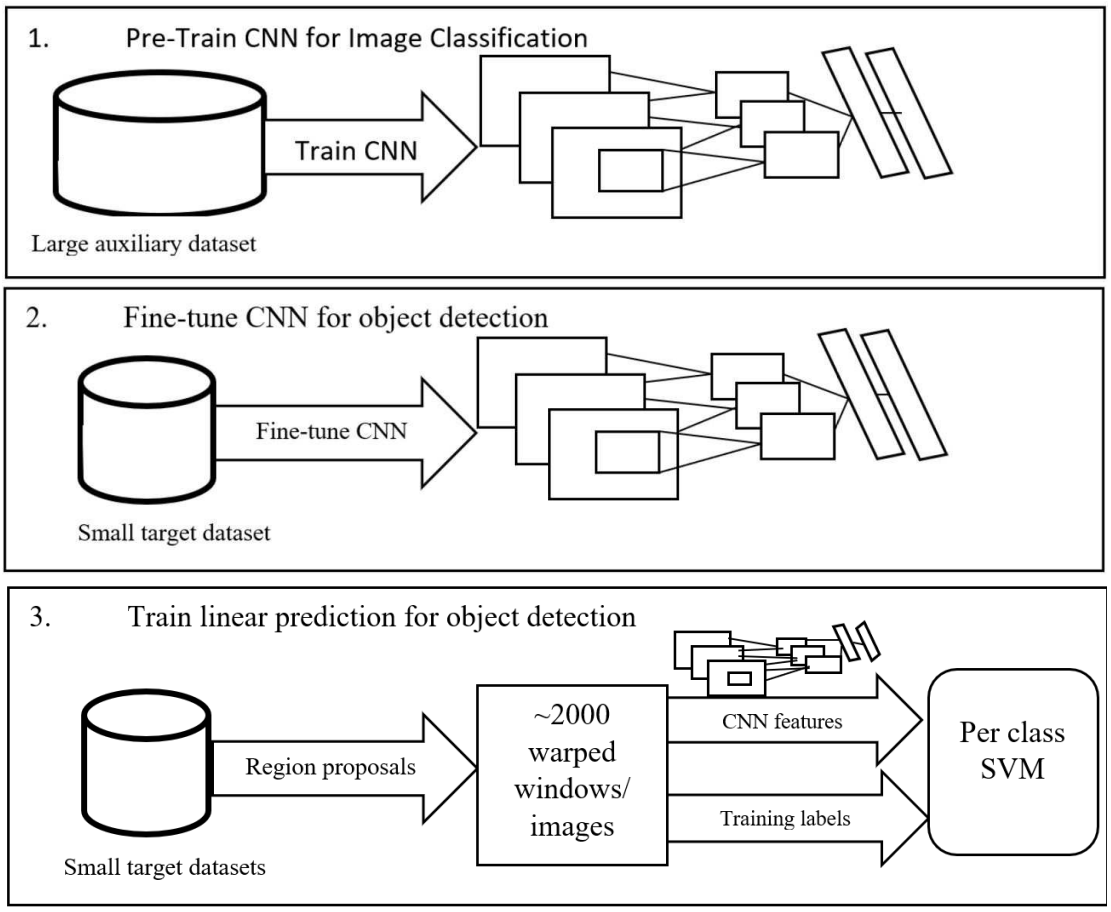


FIGURE 2. R-CNN Training (1) Pre-Train CNN for Image Classification (2) Fine-tune CNN for object detection (3) Train linear prediction for object detection

The main motive is to train convolution network which gives the coordinates of object bounding box. MultiBox loss is computed by weighted sum of confidence loss and location loss, given as:

$$\text{multibox\_loss} = \text{confidence\_loss} + \text{alpha} * \text{location\_loss},$$

where confidence and location losses are given as:

- Confidence Loss: a logistic loss on the estimates of a proposal corresponding to an object of interest.
- Location Loss: a loss corresponding to some similarity measure between the objects and the closest matching object box predictions. By default we used L2 distance.
- Alpha balances the contribution of location loss.

The parameter values are estimated in such a way to optimally reduce the loss function to make the predictions near to the ground truth.

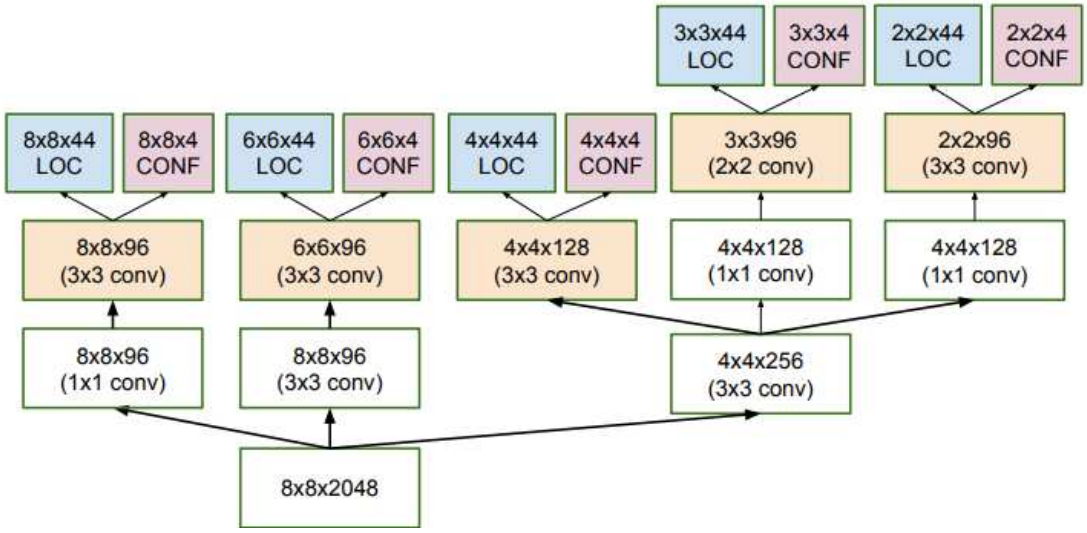


FIGURE 3. MultiBox Multiscale convolution of confidences and locations.

4.1. **MobileNets.** MobileNets is the latest approach in deep learning convolution neural network. They are really fast and small that gives results with very high precision. This approach is based on streamlined architecture which is helpful in making light weight neural network that is based on intensity distinguishable convolutions, [2].

Depthwise Separable Convolution is a type of factorised convolution, [5], that is useful in separating standard convolution into intensity wise convolution for which a single filter is applied to each input channel. Output of depthwise convolution is called pointwise convolution that is calculated by combining the outputs with 1*1 convolution. In case of standard convolution layer the input is considered as feature map I i.e. (C_I * C_I * P) and output as feature map O

i.e. (C_O * C_O * Q). Here, C_I is spatial width along with the height of a square input component feature, P shows number of input channels, C_O shows spatial width along with the height of a square output component feature, Q shows number of output channels. Then the output standard layer is parameterized by convolution kernel K which has size equivalent to C_K * C_K * P * Q. Here, D_K is the spatial coordinate of the kernel which is assumed to be a square. The output is computed as (4.1):

$$O_{k,l,n} = \sum_{i,j,m} K_{i,j,m,n} \cdot I_{k+i-1,l+j-1,m} \tag{4.1}$$

Depthwise separable convolution is made up of two layers i.e. pointwise convolutions and depthwise convolutions. Batchnorm and Rectified Linear Unit (ReLU) are the nonlinearities used for these two layers. Figure 4 shows how a standard convolution 4(a) can be reconstructed into a depthwise convolution 4(b) and a pointwise convolution 4(c).

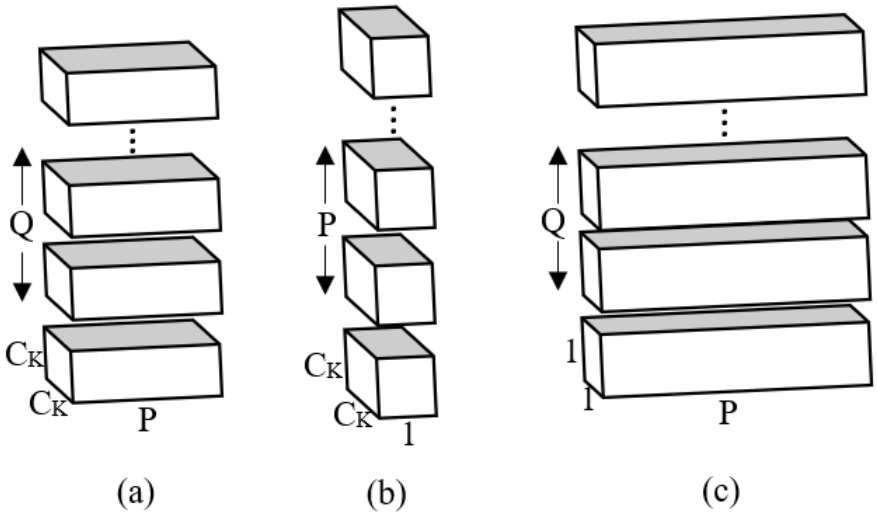


FIGURE 4. Standard Convolution Filters shown in (a) are reconstructed by two layers consisting depthwise convolution filters shown in (b) and pointwise convolution (1*1 Convolution filters) shown in (c) in context of depthwise separable filter.

For one filter per input channel depthwise convolution can be shown by the equation (4.2):

$$\widehat{O}_{k,l,n} = \sum_{i,j,m} \widehat{K}_{i,j,m,n} \cdot I_{k+i-1,l+j-1,m}, \tag{4.2}$$

where $\widehat{K}$ is the depthwise convolutional kernel having size C_1 . C_2 . P where the m^{th} filter in $\widehat{K}$ is applied to the p^{th} channel in I to produce the p^{th} channel of the filtered output feature map $\widehat{O}$.

Network Structure. All layers are trailed by a batchnorm, [10] and ReLU nonlinearity except for the last completely associated layer which has no non-linearity and goes into a softmax layer. Figure 5 shows standard convolution layer and depthwise convolutions with Batchnorm and ReLU. MobileNets has 28 layers in total, combining depthwise and pointwise convolutions as individual layers.

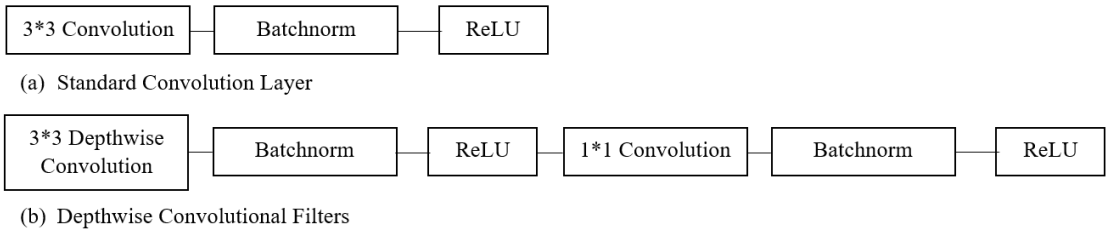


FIGURE 5. (a): Standard convolution layer with batchnorm and ReLU (b): Depthwise separable convolutions with depthwise and pointwise layers followed by batchnorm and ReLU.

Detection and Default Boxes. Every feature layer can produce a fixed model of detection prediction using a model of convolutional channels. With m channels for a feature layer of size p*q the component for finding parameters is a 3*3*m kernel that gives a point for classification. This technique is similar to YOLO, [12] which uses a midway technique of completely associated layer rather than convolution channel. For multiple feature maps of the topmost point of the system a combination of default bounding boxes with each feature cell is related. For each component level the offsets related to default boxes of the cells are predicted. Also, the per-component score that shows the closeness of the class in each of these boxes are predicted.

Data augmentation. In order to make the model powerful to multiple input objects, each processing image is randomly sampled using any of the following

methods: – Utilize the whole original input image. – Plot an area in order that the minimum jaccard overlap with the objects is between 0.1 - 0.9. – Randomly plot an area. The limit to which each sampled area is [0.1, 1] comparative to the original image size and the aspect proportion is in between 1/2 and 2. If the focal point is calculated fixed than the covered piece of the ground truth box is kept. After the testing step, each calculated fix is resized to settled size and is on a level plane flipped with likelihood of 0.5.

4.2. **Single shot multibox detector (SSD).** SSD approach is dependent on feed-forward convolutional network that yields a fixed size group of bounding boxes and calculated the precision of object classes in those boxes. Figure 6 shows the architecture of SSD MultiBox detector.

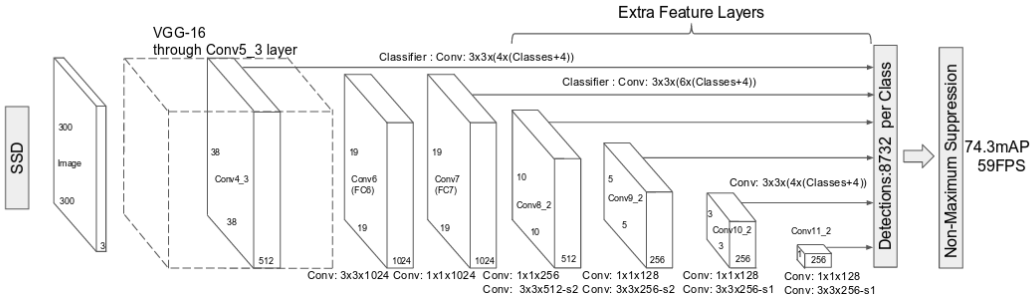
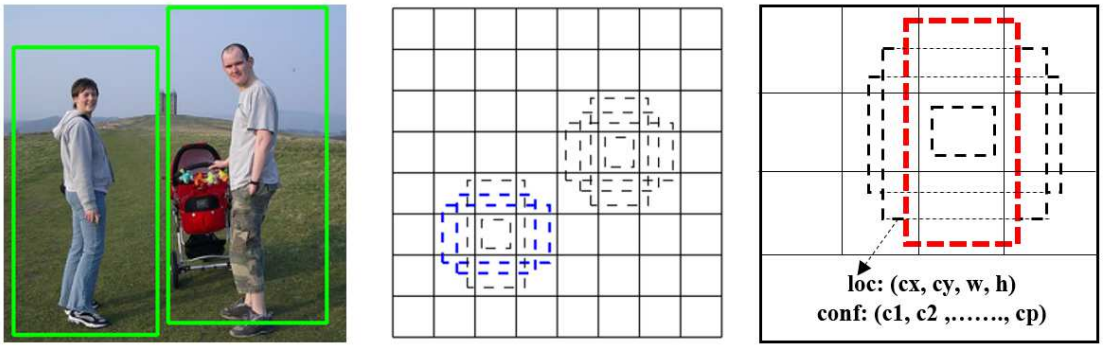


FIGURE 6. Architecture of Single Shot MultiBox Detector (input is 300*300*3).

Many improvements were done on SSD to make it more efficient for localizing and classifying objects. Fixed Priors: Each feature map is linked with a group of default bounding boxes of multiple aspect ratios and dimensions. They are chosen based on value more than 0.5 of IoU with respect to ground truth as shown in figure 7. This helps SSD to formulise for each input with pre-training. For example, assume we have calculated two diagonally opposed values (a1,b1) and (a2,b2) for each d default bounding boxes per feature map and m classes to classify on a feature map of size f= p*q, SSD computed value for this feature map is f*d*(4+m).

Location Loss: To calculate location loss, SSD uses smooth L1-Norm. It is highly effective and gives more space for operation but not as accurate as L2-Norm. This is accetable as difference of few pixels would not affect the results. Classification: SSD performs classification where as MultiBox does not.



(a)Image with GT boxes (b) 8*8 feature map (c) 4*4 feature map

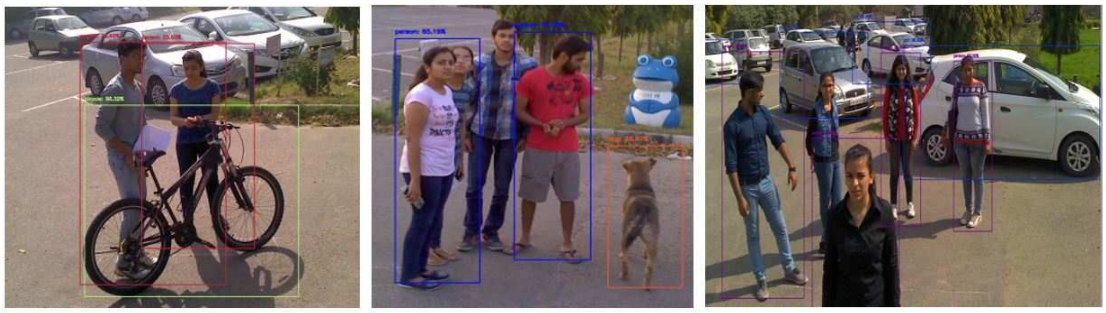
FIGURE 7. Feature Map.

Therefore, for every predicted bounding box a group of m class predictions are calculated for each class in dataset.

5. EXPERIMENTAL RESULTS

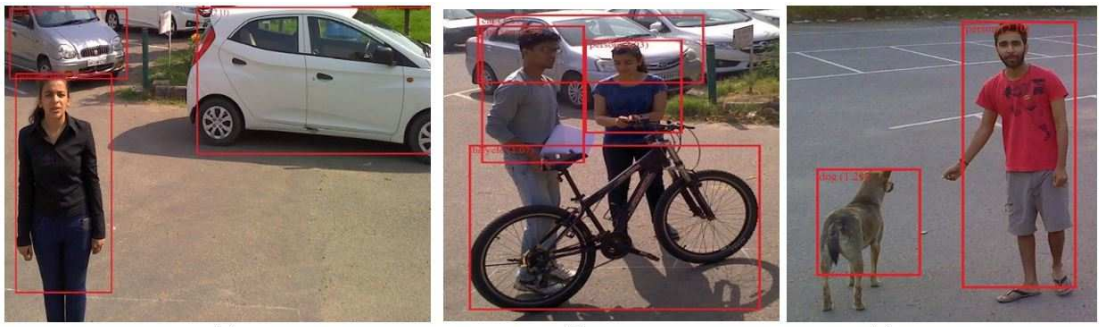
MobileNets SSD Multibox detector method can be used for super-fast, real-time object detection on resource constrained devices. This will enable us to pass input images through the network and obtain the output bounding box (x, y)-coordinates of each object in the image. Finally, we present the results of applying the MobileNet Single Shot Detector to input images and compare results with R-CNN.

5.1. **Own dataset.** In this paper we have discussed about the efficiency of Mobilenets SSD in detecting objects and now we will see its results as compared to HOG plus SVM results which is popular algorithm for pedestrian detection. We have used 145 images for testing from our dataset. When Mobilenets SSD is applied to an image it shows detected persons in bounding boxes. Bounding box boundary is coloured according to detected accuracy of objects, along with it shows the calculated confidence of the persons detected in the image. Figure 8 shows the output images when this technique is applied to own dataset on RGB images. And Figure 9 shows the output when RCNN is applied on the images. It predicts the objects with help of red coloured bounding boxes with computed accuracy.



(a) (b) (c)

FIGURE 8. (a) 2 persons detected (calculated confidence: bicycle: 98.32%, person: 76.42% and person: 25.63%) (b) 2 persons and 1 dog detected (calculated confidence: person:85.19%, person:87.51%, dog:26.67%) (c) 4 persons and 3 cars detected (calculated confidence: person:49.80%, person:90.06%, person:29.74%, person:88.68%, car:97.5%, car:95.4%, car:94.5%).



(a) (b) (c)

FIGURE 9. (a) A person and two cars are detected (b) Two person, a bicycle and a car are detected (c) A person and a dog are detected.

Confusion Matrix:

Table 1: Confusion matrix of MobileNet SSD on personal dataset.

108 (TP)	13 (TN)
23 (FN)	0 (FP)

5.2. **Pascal Voc 2012 dataset.** PASCAL VOC dataset provides standardised images for object detection and also provides a general set of tools for datasets and their annotation that allows to evaluate and compare multiple methods. PASCAL VOC 2012 has 20 classes containing 11,530 images having 6,929 segmentations and 27,450 ROI annotated objects. We have taken pedestrian images from this dataset for our results.

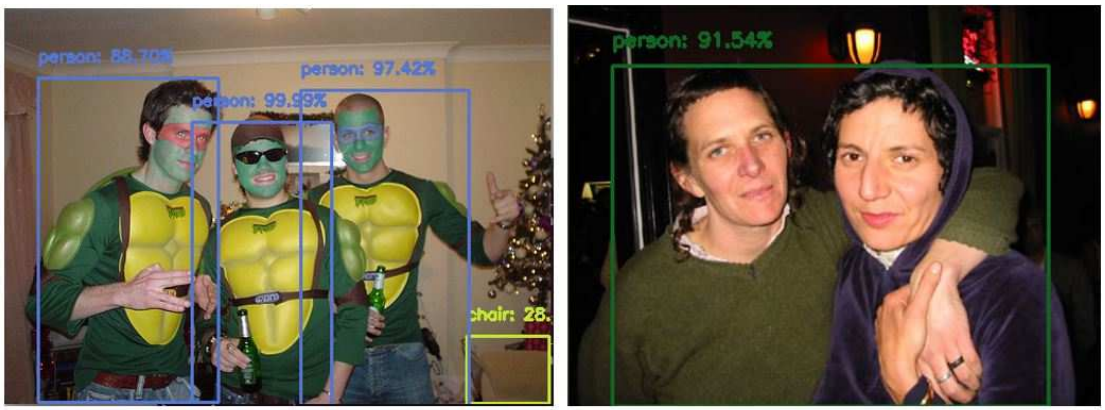


FIGURE 10. (a) 3 persons are present and are detected with 99.9%, 97.4% and 88.7% accuracy. (b) 2 persons present and 1 is detected with 91.5% accuracy.

Confusion Matrix:

Table 2: Confusion matrix of RCNN on personal dataset.

80 (TP)	10 (TN)
52 (FN)	2 (FP)

Figure 10 shows the results by applying MobileNets SSD to this dataset. Table 3 shows the resultant confusion matrix for MobileNet SSD. Figure 11 shows the results by applying RCNN to this dataset.

Confusion Matrix:

Table 3: Confusion matrix of MobileNet SSD on PASCAL VOC 2012 dataset.

126 (TP)	10 (TN)
60 (FN)	4 (FP)

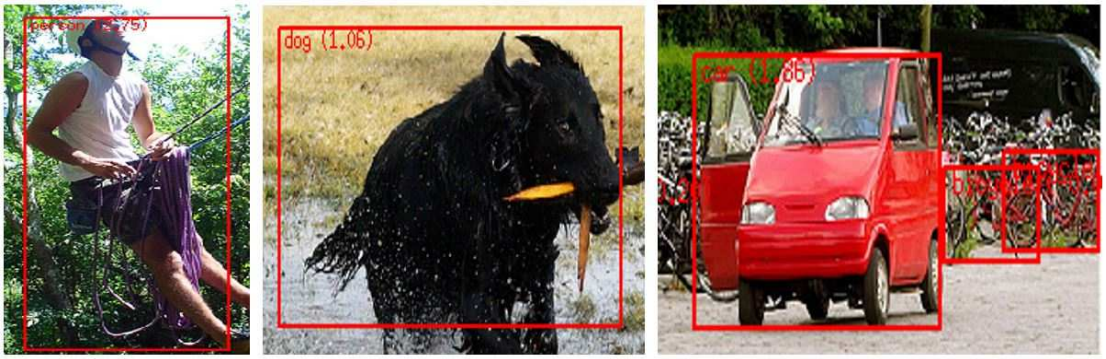


FIGURE 11. (a) 1 person is detected (b) 1 dog is detected (c) 1 car and 2 bicycles are detected.

Confusion Matrix:

Table 4: Confusion matrix of RCNN on PASCAL VOC 2012 dataset.

125 (TP)	06 (TN)
64 (FN)	05 (FP)

6. CONCLUSION

This paper compares MobileNet MultiBox SSD with RCNN. R-CNN uses region proposals with CNNs based on Selective search method. And MobileNets SSD is based on depthwise separable convolutions which uses a single filter which is applied to each input and outputs are combined by using pointwise convolution. Based on the results computed by applying techniques on same datasets, we can observe that both the techniques are accurate and fast in detecting objects whether a person, an animal or a car. MobileNets SSD is relatively faster than RCNN and also has better accuracy in terms of true positives and true negatives.

REFERENCES

[1] P. DOLLAR, C. WOJEK, B. SCHIELE, P. PERONA: *Pedestrian Detection: An Evaluation of the State of the Art*, IEEE Transactions On Pattern Analysis And Machine Intelligence, **34**(4) (2012), 743–761.

[2] J. R. UIJLINGS, K. E. VAN DE SANDE, T. GEVERS, A. W. SMEULDERS: *Selective search for object recognition*, International Journal of Computer Vision, **104**(2) (2013), 154—171.

[3] A. G. HOWARD: *Some improvements on deep convolutional neural network-based image classification*, Proceeding of IEEE Conference on Computer Vision and Pattern Recognition, 2013.

[4] J. REDMON, S. DIVVALA, R. GIRSHICK, A. FARHADI: *You only look once: Unified, real-time object detection*, Proceedings of IEEE Conference on Computer vision and Pattern recognition, 2016.

[5] P. FELZENSZWALB, D. MCALLESTER, D. RAMANAN: *A discriminatively trained, multiscale, deformable part model*, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, 2008, 1–8.

[6] R. GIRSHICK, J. DONAHUE, T. DARRELL, J. MALIK: *Rich feature hierarchies for accurate object detection and semantic segmentation*, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, 580–587.

[7] P. DOLLAR, Z. TU, P. PERONA, S. BELONGIE: *Integral channel features*, Proceedings of British Machine Vision Conference, 2009, 91.1–91.11.

[8] P. VIOLA, M. J. JONES: *Robust real-time face detection*, International Journal of Computer Vision, **57**(2) (2004), 137–154.

[9] P. DOLLAR, R. APPEL, S. BELONGIE, P. PERONA: *Fast feature pyramids for object detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **36**(8) (2014), 1532–1545.

[10] W. NAM, P. DOLLAR, J. H. HAN: *Local decorrelation for improved pedestrian detection*, Proceedings of the 27th International Conference on Neural Information Processing Systems, **1** (2014), 424–432.

[11] R. BENENSON, M. OMRAN, J. HOSANG, B. SCHIELE: *Ten years of pedestrian detection, what have we learned?*, European Conference on Computer Vision workshop, Part of the Lecture Notes in Computer Science book series, **8926** (2014), 613–627.

[12] R. GIRSHICK, J. DONAHUE, T. DARRELL, J. MALIK: *Rich feature hierarchies for accurate object detection and semantic segmentation*, IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, 580–587.

[13] Y. TIAN, P. LUO, X. WANG, X. TANG: *Pedestrian detection aided by deep learning semantic tasks*, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, 2015, 5079–5087.

[14] Y. TIAN, P. LUO, X. WANG, X. TANG: *Deep learning strong parts for pedestrian detection*, IEEE International Conference on Computer Vision (ICCV), Santiago, 2015, 1904–1912.

[15] P. DOLLAR, R. APPEL, S. BELONGIE, P. PERONA: *Fast feature pyramids for object detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **36**(8) (2014), 1532–1545.

[16] R. GIRSHICK, J. DONAHUE, T. DARRELL, J. MALIK: *Rich feature hierarchies for accurate object detection and semantic segmentation*, IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, 580-587.

[17] K. HE, X. ZHANG, S. REN, J. SUN: *Spatial pyramid pooling in deep convolutional networks for visual recognition*, Computer Vision – ECCV 2014, Lecture Notes in Computer Science, **8691** (2014), 346–361.

[18] R. GIRSHICK: *Fast R-CNN*, IEEE International Conference on Computer Vision (ICCV), Santiago, 2015, 1440–1448.

[19] D. ERHAN, C. SZEGEDY, A. TOSHEV, D. ANGUELOV: *Scalable object detection using deep neural networks*, IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, 2155–2162.

[20] S. REN, K. HE, R. GIRSHICK, J. SUN: *AFaster R-CNN: Towards real-time object detection with region proposal networks*, IEEE Transactions on Pattern Analysis and Machine Learning, **39**(6) (2017), 1137–1149.

[21] A. SHRIVASTAVA, A. GUPTA, R. GIRSHICK: *Training region-based object detectors with online hard example mining*, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, 761–769.

[22] C. SZEGEDY, S. REED, D. ERHAN, D. ANGUELOV, S. IOFFE: *Scalable, High-Quality Object Detection*, Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, 2147–2154.

[23] P. SERMANET, D. EIGEN, X. ZHANG, M. MATHIEU, R. FERGUS, Y. LECUN: *Overfeat: Integrated recognition, localization and detection using convolutional networks*, Proceedings of the IEEE conference on computer vision and pattern recognition, 2014.

[24] L. SIFRE: *Rigid-motion scattering for image classification*, Ph. D. thesis, 2014.

[25] S. IOFFE, C. SZEGEDY: *Batch normalization: Accelerating deep network training by reducing internal covariate shift*, ICML'15: Proceedings of the 32nd International Conference on International Conference on Machine Learning, **37** (2015), 448–456.

[26] J. JIN, A. DUNDAR, E. CULURCIELLO: *Flattened convolutional neural networks for feedforward acceleration*, International Conference on Learning Representations, 2014.

[27] F. N. IANDOLA, M. W. MOSKEWICZ, K. ASHRAF, S. HAN, W. J. DALLY, K. KEUTZER: *Squeezenet: Alexnet-level accuracy with 50x fewer parameters andi 1mb model size*, ICLR, 2017.

[28] M. JADERBERG, A. VEDALDI, A. ZISSERMAN: *Speeding up convolutional neural networks with low rank expansions*, Proceedings of British Machine Vision Conference, 2014.

[29] V. LEBEDEV, Y. GANIN, M. RAKHUBA, I. OSELEDETS, V. LEMPITSKY: *Speeding-up convolutional neural networks using fine-tuned cp-decomposition*, ICLR, 2015.

[30] G. HINTON, O. VINYALS, J. DEAN: *Distilling the knowledge in a neural network*, NIPS 2014 Deep Learning Workshop, 2015.

[31] X. WANG, M. YANG, S. ZHU, Y. LIN: *Regionlets for generic object detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **37**(10) (2013), 2071–2084.

[32] A. KRIZHEVSKY, I. SUTSKEVER, G. HINTON: *ImageNet classification with deep convolutional neural networks*, Proceedings of Neural Information Processing Systems, **60**(6) (2012), 84–90.

[33] I. ENDRES, D. HOIEM: *Category independent object proposals*, Proceedings of European Conference on Computer Vision, 2010, 575–588.

[34] D. BISWAS, H. SU, C. WANG, A. STEVANOVIC: *An automatic traffic density estimation using Single Shot Detection (SSD) and MobileNet-SSD*, Elsevier Journal Physics and chemistry of Earth, **110** (2019), 176–184.

[35] Y. PI, N. D. NATH, A. H. BEHZADAN: *Convolutional neural networks for object detection in aerial imagery for disaster response and recovery*, Elsevier Journal Advanced Engineering Informatics, **43** (2020), 101009.

DEPARTMENT OF INFORMATION AND TECHNOLOGY
UIET, PANJAB UNIVERSITY
Email address: khushaboogill@gmail.com

DEPARTMENT OF INFORMATION AND TECHNOLOGY
UIET, PANJAB UNIVERSITY
Email address: veenumangat@yahoo.com