IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-2, Feb 2019 E-ISSN: 2347-2693

Real Time Object Detection Can be Embedded on Low Powered Devices

Jimut Bahan Pal[1*], Shalabh Agarwal[2]

[1,2]Dept. Of Computer Science, St. Xavier's College, Park Street, Kolkata, India

[*]*Corresponding Author: jimutbahanpal@yahoo.com, Tel.: +00-12345-54321*

DOI: https://doi.org/10.26438/ijcse/v7i2.10051009 | Available online at: www.ijcseonline.org

Accepted: 26/Feb/2019, Published: 28/Feb/2019

***Abstract*—** It has been a real challenge for computers with low computing power and memory to detect objects in real time. After the invention of Convolution Neural Networks (CNN) it is easy for computers to detect images and recognize them. There are several technologies and models which can detect objects in real time, but most of them require high end technologies in terms of GPUs and TPUs. Though, recently many new algorithms and models have been proposed, which runs on low resources. In this paper we studied MobileNets to detect objects using webcam to successfully build a real time object detection system. We observed the pre trained model of the famous MS COCO dataset to achieve our purpose. Moreover, we applied Google's open source TensorFlow as our back end. This real time object detection system may help in future to solve various complex vision problems.

***Keywords*—** TensorFlow, MobileNet, MS COCO, Real-time, and Object detection.

I. INTRODUCTION

It is too easy for humans to perceive objects and tell about their location, what the objects are, number of them, how they interact with each other etc. The animals' visual system is too complex which costs little conscious thought for them to eat, manoeuvre, and even do complex tasks easily. The scope of image recognition and real time object detection is vast. It may be used by self-driving cars; it may be used in real time surveillance, proposed by Tripathi et al. **[1]**. It can also be used to help people who cannot see properly by describing what an image is through Natural Language Processing (NLP). We may also process real time data such as traffic signalling by checking the traffic density and then controlling the traffic signals, or maybe to help generic robots to do specific tasks and many more.

Li et al.**[2]** studied that the field of object detection in machine learning is considered as one of the challenging tasks in real life since it involves high floating point computation, object classification and localization. To be precise some methods classify the proposal regions into object categories and recent methods generally unify localization and classification stages. The present object detection algorithms scale an image in context, use classifier at various parts of the image by making bounding boxes and finally predicts about the image. The recent methods are very resource hungry which requires large memory and model size to work. Hence, it takes more computing power and consumes more electrical power. Thus, these are basically inefficient for embedded system architectures.

II. RELATED WORK

Several models have been proposed recently which solves the problem of object detection; some are real time needing less computing power, while others require high computing power and GPUs to detect from the input. The state of the art object detection models is Convolutional Neural Networks (CNN). It generally uses lot of resources in classifying objects. Early methods **[2]** such as R-CNN and Fast-RCNN, divided the input images to bounding boxes, then classify on each bounded boxes. One convolutional network extract features **[3]** and a linear model adjust the bounding boxes. Here, a non max suppressor eliminates duplicate deductions of the boxes from the image. Tripathi et al. **[1]** implemented a low complexity fully convolutional neural network named LCDet, which can work on embedded systems. They trained the model on publicly available FDDB and Wideface dataset which contains a number of varying faces, and they finally developed a real time system which performs on low embedded system architectures as shown in **Fig. 1, Fig. 2, Fig. 3 and Fig. 4**.

Fig. 1: Faces detected in black and white images on FDDB. **Picture Courtesy:** Subarna Tripathi, UC San Diego **[1]**

Fig. 2: Multiple faces of frontal and side profiles on FDDB. **Picture Courtesy:** Subarna Tripathi, UC San Diego **[1]**

Fig. 3: Successful face detection results of LCDet on challenging Widerface validation images containing complex examples **Picture Courtesy:** Subarna Tripathi, UC San Diego **[1]**

Zhang et al.**[4]**designed ShuffleNet as shown in **Fig. 5**, which is an extremely efficient CNN for mobile devices. It highly reduces computation by doing point wise group convolution and channel shuffle. It maintains accuracy at the same time. They have experimented on MS COCO and ImageNet dataset which showed that it is superior to other models both in terms of computation and complexity.

Wang et al. **[5]** designed a model, named Pelee, which outperformed MobileNets, NASNet-A, and Shuffle Net. Since they depend on depth wise separable convolution, hence they are inefficient compared to Pelee. Though, Pelee is 66% of the Mobile Net's model size, but it outperformed. Li et al. **[2]** introduced Tiny-DSOD: Lightweight object detection for Resource Restricted Usages. It has two blocks namely depth wise dense block (DDB) and depth wise feature pyramid network (D-FPN) based front end. It outperforms Tiny-YOLO, MobileNet-SSD (v1 & v2), SqueezeDet, Pelee etc. on all three benchmarks (PASCAL VOC 2007, KITTI, MS COCO).

Fig. 4: Face detected on Widerface challenge by LCDet. Yellow faces are called as false positives as per 50% IoU criteria. The regions marked in red show missed detections **Picture Courtesy:** SubarnaTripathi, UC San Diego **[1]**

In recent years the smart surveillance technologies have evolved, which uses AI and ML technologies, along with intensive computing resources that cannot be embedded into low computing chips. Nikouei et al. **[6]** devised Kerman (Kernelized Kalman Filter): A hybrid lightweight tracking algorithm to enable smart surveillance as an edge service. Decision tree based on hybrid Kernelized Correlation Filter (KCF) in Kerman, in addition to a very lightweight Convolutional Neural Network (L-CNN), helps it to track object of interest in a very low computing resource environment with a decent accuracy.

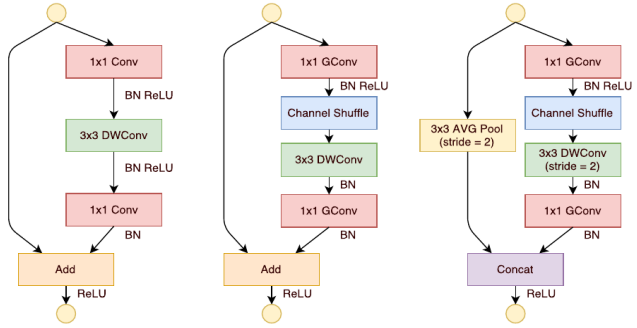

Fig. 5: ShuffleNet Units, left (bottleneck unit with depth wise convolution); middle (ShuffleNet unit with point wise group convolution and channel Shuffle); right (ShuffleNet unit with stride = 2) **Picture Courtesy:** Xiangyu Zhang, MegviiInc (Face++) **[4]**

Liu et al. **[7]** built SAM – RCNN: Scale Aware Multi-Channel pedestrian detection, which can generically select multi-resolution convolutional features according to the resolution and pedestrian sizes. It is better than the classical counter parts which assembles features from multiple layers of CNN, but doesn't guarantee the optimisation of the feature representation. They proved the superiority of their proposed method by evaluating it on the KITTY and Caltech benchmarks.

Redmon et al. **[3]** have proposed You Only Look Once (YOLO): Unified, Real Time Object Detection. They framed the classification problem as a regression problem, to spatially separated bounding boxes and associated class probabilities, using a single neural network to predict directly from full images in one evaluation. It is extremely scalable and can be optimized as it works on a single network. It is superior to DPM and RCNN in all fields.

III. METHODOLOGY

We used MS COCO dataset **[8]**, MobileNets **[9]** and TensorFlow **[10]** API to build our model. TensorFlow is used to implement and execute Machine Learning algorithms developed by Abadi et al. **[10]**. An algorithm using TensorFlow can be executed in a variety of systems ranging from small scale low powered mobile devices to a large scale distributed systems of thousands of GPU cards. It has been developed and built at Google. It is used for a variety of purposes in the field of computer science like computer vision, speech recognition, robotics, geographic information retrieval and extraction, NLP, and many more.

The MobileNets model is based on depthwise separable convolutions. It divides a standard convolution into a depthwise convolution, **1x1** convolutions that are called as point wise convolution. A single filter is applied to each input channel through depthwise convolution in MobileNets. The depth wise convolution is fed by the outputs of **1x1** point wise convolution. The input is then filtered and combined by a standard convolution in one step to a new set of outputs. This is split into two layers by the depthwise separable convolution, one for filtering and another for combining, which drastically decreases the model size and computational cost.

The first layer of MobileNets is fully convolutional. The network is defined in simple terms so that we can easily explore its topology and find a good network. It follows a series of batch norm or ReLU layers. The final layer is exceptional; it doesn't have nonlinearity and feeds into a softmax layer for classification.

In this investigation we used pre-trained model using the Microsoft COCO: Common Objects in Context dataset **[8]**. The dataset has 91 common objects as categories with 82 of them having more than 5000 labeled instances. It has 2,500,000 labelled instances in total of 328,000 images as shown in **Fig. 6** and **Fig. 7**. It has more instances per category as compared to the ImageNet. This thing helps in learning detailed object models which is capable of precise 2D localization.

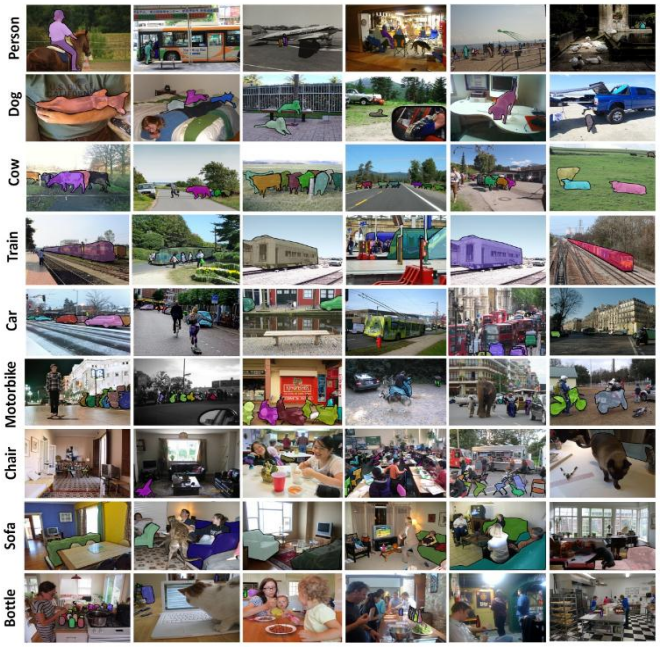

Fig. 6: Samples of annotated images in the MS COCO dataset **Picture Courtesy:** Tsung-Yi Lin, Cornell NYC Tech **[8]**

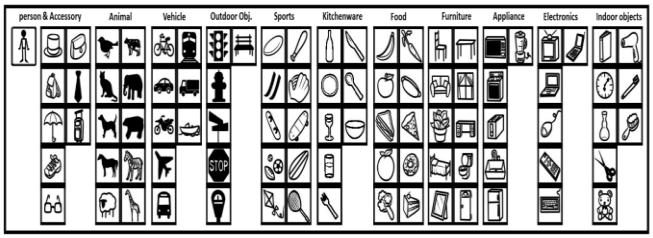

Fig. 7: Icons of 91 categories in the MS COCO dataset grouped by 11 super-categories. **Picture Courtesy:** Tsung-Yi Lin, Cornell NYC Tech **[8]**

IV. RESULTS AND DISCUSSION

We successfully devised our model on MobileNet **[9],** which runs on low power devices without any GPU. It can also run on mobile and embedded vision applications. MobileNet uses depth-wise convolution to build light weight deep neural networks and are based on streamline architecture. It can successfully classify 91 different objects. Some of the known and easily available real world objects has been easily detected and shown in **Fig. 8**. Here, we used a low powered laptop with no external GPU to classify these images and we found our desired results; the objects were detected easily using Laptop's webcam. The input video through the webcam was hazy, but it was successfully detected.

There has been rising interest in the scientific community to build small and efficient neural networks that works on low powered devices. They can be either thought of as by training small networks or by compressing pre-trained network. We can improve this model to do various effective

things through it; for example, Espinosa et al. **[11]** have studied and designed a real time motorcycle detection system. Due to increase in number of vehicles in urban areas there has been rise in accident and fatality rates, therefore, this has been an important part of their discovery. They used a Faster R-CNN model for real time detection and classification of motorcycle, which can deal with highly occluded image and achieve a precision of 75% setting up a benchmark in image recognition task.

It is also unlikely that pedestrians will always carry a pedestrian safety hand held device with them and get signals through the help of a low latency communication device. We can improve this model as developed by Rahman et al. **[12]** where they studied the model with the help of traffic camera to precisely pinpoint the location of pedestrians using deep learning to broadcast safety alerts related to pedestrians to nearby vehicles around signalized intersections. They successfully build an efficient real time pedestrian detection strategy combining pedestrian algorithm using deep learning while maintaining high object detection accuracy.

Fig. 8: The real time images detected using MobileNetthrough web-cam

The model we studied using MobileNet allows developers to specifically choose a small network, which matches the computational power of the devices and is flexible and scalable. We can scale the latency and size of the network for any application. In this way we successfully build real time object detection algorithms on small computing devices, saving power and computational costs.

V. CONCLUSION AND FUTURE SCOPE

We have successfully built a real time object detection system that can detect up to 91 varieties of objects. In this study TensorFlow and MobileNetis used as backend and models for this investigation. This model may be improved both in terms of accuracy, speed and complexity. There have been several ongoing projects which help to improve quality of life and makes living of human beings easier. The real time object detection was once an impossible task, but now after the invention of CNN it has been one of the most demanding topics in the history of computer vision. The main aim of object detection is to build a full scale system which can detect any object in the world and describes information and behaviour of the object as soon as it sees the object. This is the beginning of highly intelligent computer vision algorithms which can work on low computing system. We will be developing and designing this model more using some newer algorithms to improve its accuracy in future. Finally, we conclude that it is possible to detect objects in real time using MobileNets.

ACKNOWLEDGMENT

I greatly appreciate the help received from my father Dr. Jadab Kumar Pal, Deputy Chief Executive, Indian Statistical Institute, Kolkata, for constantly help and support to develop this model. I am thankful to Mr. Aniket Bhattacharyea, for his inspiration. I sincerely acknowledge the motivation and support received from Dr. AsokeNath, Professor in the Department of Computer Science, St. Xavier's College, Kolkata for helping me with the design of the paper and providing his constant support in every minute detail. I also appreciate the help received from Xiangyu Zhang (Microsoft Research), SubarnaTripathi (UC San Diego) and Tsung-Yi Lin from Cornell NYC Tech for allowing me to use their pictures in my research paper for demonstration.

REFERENCES

[1] S. Tripathi, G. Dane, B. Kang, V. Bhaskaran, and T. Nguyen, "LCDet: Low-Complexity Fully-Convolutional Neural Networks for Object Detection in Embedded Systems", work done in part during an internship at Qualcomm, arXiv:1705.05922 [cs.CV], https://arxiv.org/abs/1705.05922 accessed on 27.09.2018, May 2017.

[2] Y. Li, J. Li, W. Lin, and J. Li, "Tiny-DSOD: Lightweight Object Detection for Resource-Restricted Usages", Shanghai Jiao Tong University and Intel Labs, arXiv: 1807.11013 [cs.CV], https://arxiv.org/abs/1807.11013 accessed on 27.09.2018, July 2018.

[3] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", arXiv: 1506.02640 [cs.CV], https://arxiv.org/abs/1506.02640 accessed on 27.09.2018, June 2015.

[4] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices", MegviiInc (Face++), arXiv:1707.01083 [cs.CV], https://arxiv.org/abs/1707.01083 accessed on 27.09.2018, Dec 2017.

[5] R. J. Wang, X. Li, S. Ao, and C. X. Ling, "Pelee: A Real-Time Object Detection System on Mobile Devices", University of

Western Ontario, arXiv: 1804.06882v1 [cs.CV], https://arxiv.org/abs/1804.06882 accessed on 27.09.2018, April 2018.

[6] S. Y. Nikouei, Y. Chen, S. Song, and T. R. Faughnan, "Kerman: A Hybrid Lightweight Tracking Algorithm to Enable Smart Surveillance as an Edge Service", Binghamton University, arXiv: 1808.02134 [cs.DC], https://arxiv.org/abs/1808.02134 accessed on 27.09.2018, August 2018.

[7] T. Liu, M. Elmikaty, and T. Stathaki, "SAM-RCNN: Scale-Aware Multi-Resolution Multi-Channel Pedestrian Detection", Electrical and Electronic Engineering Imperial College London and Jaguar Land Rover Research Coventy, arXiv: 1808.02246 [cs.CV], https://arxiv.org/abs/1808.02246 accessed on 27.09.2018, August 2018.

[8] T. Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollar, "Microsoft COCO: Common Objects in Context", arXiv: 1405.0312 [cs.CV], https://arxiv.org/abs/1405.0312 accessed on 27.09.2018, May 2014.

[9] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications", arXiv: 1704.04861 [cs.CV], https://arxiv.org/abs/ 1704.04861 accessed on 27.09.2018, April 2017.

[10] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Yuan Yu, X. Zheng ,"TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems", Google Research, available on http://download.tensorflow.org/paper/whitepaper2015.pdf accessed on 27.09.2018, November 2015.

[11] J. E. Espinosa, S. A. Velastin, and J. W. Branch, "Motorcycle detection and classification in urban Scenarios using a model based on Faster R-CNN", University Carlos 3 – Madrid Spain, arXiv: 1808.02299 [cs.CV], https://arxiv.org/abs/1808.02299 accessed on 27.09.2018, August 2018.

[12] M. Rahman, M. Islam, J. Calhoun, and M. Chowdhury,"Real-time Pedestrian Detection Approach with an Efficient Data Communication Bandwidth Strategy", Clemson University, arXiv: 1808.09023 [cs.CV], https://arxiv.org/abs/1808.09023 accessed on 27.09.2018, August 2018.

Authors Profile

Jimut Bahan Pal is a student of Dept. of Computer Science, St. Xavier's College (Autonomous), Kolkata-700016. He is fluent in Python programming language, and has invented various scraping programs. He is also a member of various online learning communities like Coursera, Stanford Online, edX etc. Being a Machine Learning enthusiast, he has done various minor projects in Machine Learning area. He has also built various RPG games with Unity-3D, and Unity-2D game engines in association with MOOCs. He is an aspiring and curious researcher.

Shalabh Agarwal is the Head of the Department of Computer Science and Systems-in-charge of St. Xavier's College [Autonomous], Kolkata. He is also the Director of Computer Centre and Central Computing Facilities of the College. He is the recipient of many International and National awards for his contribution in computer education and research.

