DOI: 10.15849/IJASCA.211128.10

Int. J. Advance Soft Compu. Appl, Vol. 12, No. 3, November 2021
ISSN 2074-8523; Copyright © ICSRS Publication, 2021
www.i-csrs.org

Deep Learning for the Improvement of Object Detection in Augmented Reality

Zainab Oufqir, Lamiae Binan, Abdellatif EL ABDERRAHMANI and Khalid Satori

LISAC,Department of Mathematics and informatic University Sidi Mohamed Ben Abdellah, Faculty of Sciences Fes, Morocco
zainab.oufkir@usmba.ac.ma, lamiaebinan@gmail.com,
abdellatif.elabderrahmani@usmba.ac.ma, khalid.satori@usmba.ac.ma

Abstract

In this article, we give a comprehensive overview of recent methods in object detection using deep learning and their uses in augmented reality. The objective is to present a complete understanding of these algorithms and how augmented reality functions and services can be improved by integrating these methods. We discuss in detail the different characteristics of each approach and their influence on real-time detection performance. Experimental analyses are provided to compare the performance of each method and make meaningful conclusions for their use in augmented reality. Two-stage detectors generally provide better detection performance, while single-stage detectors are significantly more time efficient and more applicable to real-time object detection. Finally, we discuss several future directions to facilitate and stimulate future research on object detection in augmented reality.

Keywords: *object detection, deep learning, convolutional neural network, augmented reality.*

1 Introduction

Object detection is one of the most studied issues in augmented reality [1], it is the most important step in the pose calculation to correctly align a virtual object in the real world [2]. Object detection is a computer vision technique that identifies and locates a certain object in an image of a scene [3]. Deep learning is often used for object detection [4], this detection is done in two steps: image classification and image localization. Image classification recognizes objects in the image, such as cars or people. Image localization provides the specific location of these objects

[5]. Convolutional Neural Network (CNN) is a deep learning algorithm and the most powerful type of neural network for image classification [6], the network takes an input image, detects features and classifies the detected objects into certain categories (e.g. car, truck, motor). Due to the rapid evolution of technology, the creation of new algorithms and architectures, and the exponential increase in the volume of data, the efficiency of deep learning has been improved. Thus, the use of deep learning has improved the development of more efficient, interactive and intelligent applications [7] [8] [9]. Many tasks can now be solved efficiently and with great precision using deep neural networks, sometimes even exceeding human performance. Object detection covers a variety of applications, including robotic vision [10], security [11], autonomous driving [12], human-computer interaction [13], content-based image retrieval [14], smart video surveillance [15] and augmented reality [16]. The new object detection algorithms based on deep learning are markerless methods [17], they are divided into two categories, two-stage detectors such as RCNN [18], Fast RCNN [19] and FASTER RCNN [20]. On the other hand, we have single-stage detectors, such as YOLO [21] et SSD [22]. In the next section, we detail these algorithms, see their different functionalities, their advantages and disadvantages. The experimentation part presents different tests on single-stage detectors to evaluate their performance in real time and then interpret the obtained results.

2 Two-stage detectors

Two-stage detectors like R-CNN, Fast R-CNN and Faster R-CNN use a region proposal network where the processed image is converted to feature maps to generate regions of interest in the first stage, then sends the region proposals into region classifiers that predict the category of the proposed region. These models reach the best accuracy rates, but are slower.

2.1 RCNN

R-CNN is the first object detectors based on deep learning and are an example of a two-stage detector. R-CNN uses the selective search to extract only 2000 regions of the image called by region proposals using the selective search algorithm. It then uses a pre-formed AlexNet classification model to extract a feature vector 4096 for each region. Finally, it classifies each region using the SVM and, based on the results, refines the CNN for detection, Figure 1 shows the operation of RCNN.

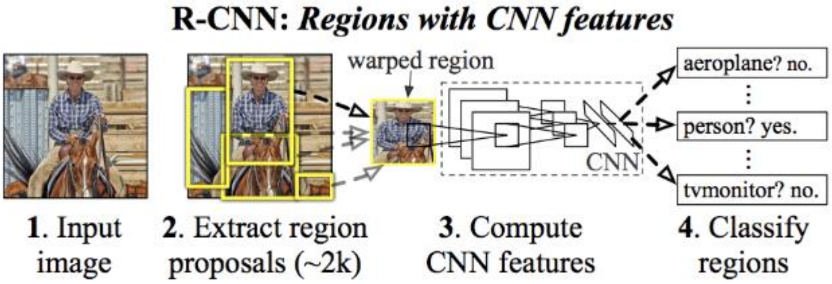


Fig1. RCNN workflow

The various problems encountered with RCNN are as follows:

- Training on the SVM classifier and bounding box regressor are both expensive in disk space and time, as the CNN feature has to be extracted from every object proposal in every image, which causes great challenges for large-scale detection.
- Training the network consumes a lot of time, because it is necessary to classify 2000 region proposals by image.
- It cannot be implemented in real time, it takes about 47 seconds for each test image due to the region proposal algorithm.
- Tests are slow because CNN functionality is extracted by object proposition in each test image, without shared computation.

2.2 Fast RCNN

The Fast RCNN corrects some of the problems in RCNN by improving the speed and quality of detection. Instead of applying CNN 2000 times to the proposed areas, it only transfers the original image of the pretrained CNN model once. The search for the selective algorithm is calculated based on the output map of the characteristics of the previous step. The ROI pool level is then used to provide a standard and predefined output size. These valid outputs are passed to the fully connected level as inputs. Finally, the two output vectors are used to predict the observed object using a softmax classifier and adapt the localization of the bounding box using a linear regressor:

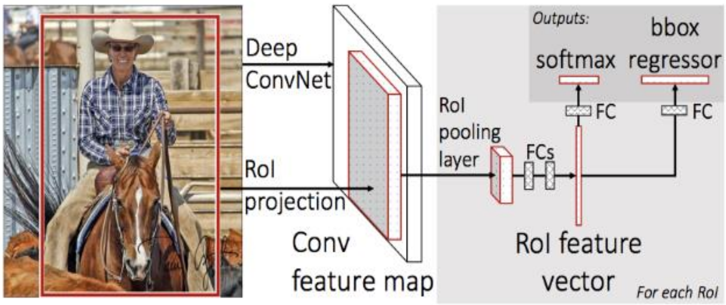


Fig2. Fast RCNN workflow

- In the figure 2, the content in the red box in the first figure is the sentence extracted, the middle part is the conv convolution feature map obtained after deep convolution, and the gray part in the figure is the proposal region in the red box corresponds to the position in the conv feature map, and then this feature is processed by the ROI association layer and then fully connected. The ROI feature vector obtained here is finally used, one is used for softmax regression after full connectivity for classification, and the other is used for box regression after full connectivity

The various problems encountered with Fast RCNN are as follows:

- Most of the time taken by Fast R-CNN is during the detection of region proposal generation by selective search.

2.3 FASTER RCNN

Faster RCNN goes further than Fast R-CNN. The search for the electoral process is being replaced by the Regional Proposals Network (RPN). The RPN network is used to create the region's offerings. This layer specifies the Softmax to get positive or negative, and then use bounding box regression, fixes the anchors to get accurate sentences. ROI Pool collects the input and offer maps, and extracts the function maps after integrating this information and feeds the subsequent full connective layer to define the target category. Finally, the classification. Use the proposal function maps to calculate the proposal category and then the bounding box regression gets the final exact location of the detection block as shown in Figure 3.

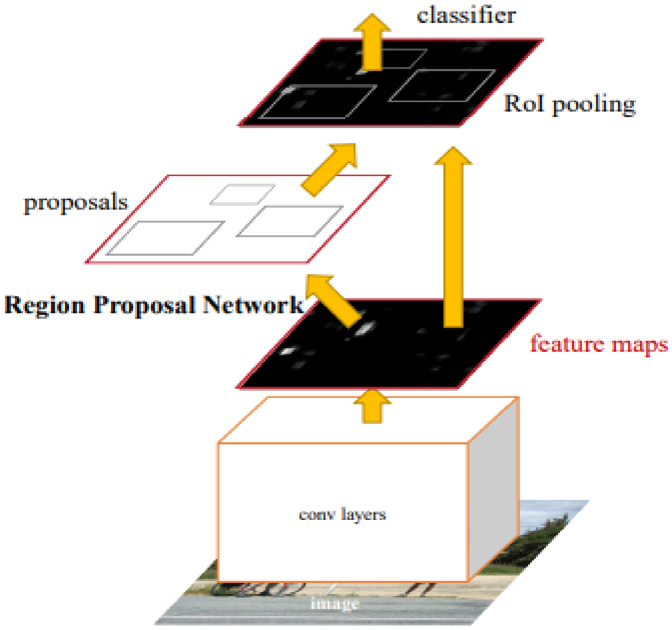


Fig3. Faster RCNN workflow

The various problems encountered with Faster RCNN are as follows:

- The object proposal takes time and as there are different systems working one after the other, the performance of the systems depends on how the previous system worked.

2.4 Comparaison

After the appearance of R-CNN, a large number of improved models have been proposed, Fast R-CNN optimizes classification and regression tasks by bounding boxes, Faster R-CNN uses an additional subnetwork to generate region proposals. The following table summarizes the performance of each detector [18] [23] :

Table 1: Comparison between RCNN, FAST RCNN and FASTER RCNN

	RCNN	**Fast RCNN**	**FASTER RCNN**
Method for generating region proposals	Selective Search	Selective Search	Region Proposal Network
The map on Pascal VOC 2007	66 mAP	66.9 mAP	66.9 mAP
Detection time (seconde)	50	2	0.2

The two-stage detectors shown use regions to identify objects. The network does not look at the whole image at once, it focuses on parts of the image sequentially. This creates some complications:

- The algorithm requires many passes in a single image to extract all the objects
- Since different systems operate one after the other, the performance of more advanced systems depends on the performance of previous systems.
- The major disadvantage of two-stage detectors is the calculation time, which is still not adapted to real time, which is why the single-stage methods have been developed.

3 Single-stage detectors

One-step detectors such as YOLO (You Only Look Once) and SSD (Singe Shot MultiBox Detector) handle object detection as a simple regression problem by taking an input image and learning the class probabilities and selection frame coordinates. They achieve lower accuracy rates, but they are much faster than two-stage object detectors.

3.1 Yolo

YOLO stands for "You Only Watch One Time: Unified Real-Time Object Detection.". The main idea is to transform target detection into regression problem solving based on a separate end-to-end network. Complete the input of the original image to output the position and category of the object. The YOLO workflow is divided into different processes, first step divides the original image into an SxS grid. If the center of the target hits a specific grid, that grid is responsible for target detection. Each grid must predict the bounding box B and the probabilities of the class C, each bounding box must predict a confidence value. Since the input image is divided into SxS grids, each grid includes 5 predictors: (x, y, w, h, confidence) and class C.

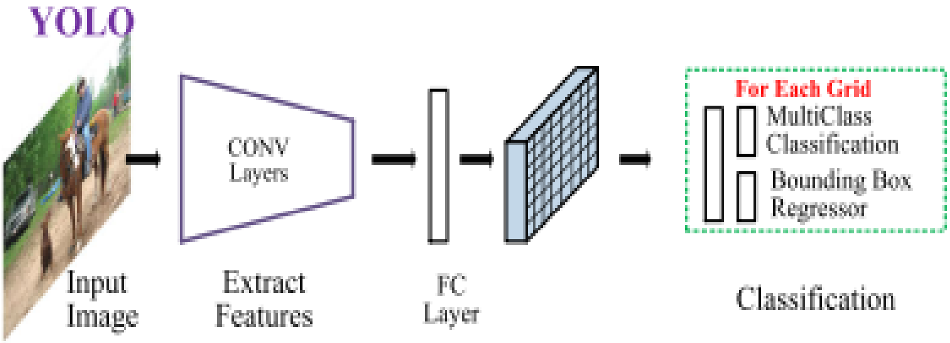


Fig4. Yolo workflow

YOLO make a few location errors. In addition, the recall rate of YOLO data is relatively low. Thus, in the second version of YOLO [24], they focused primarily on improving recall and localization while maintaining the accuracy of classification for better performance.
YOLOv3 [25] predicts a confidence score for each bounding box using logistic regression, while YOLO and YOLOv2 use the sum of squared errors for the classification terms. Linear regression of the prediction of the offset leads to a decrease in the mAP.

3.2 SDD

SSD (Singe Shot MultiBox Detector) is a single shot detector for multiple categories, its faster than the previous state of the art for single shot detectors (YOLO) and more accurate, in fact as accurate as slower techniques that perform explicit and pooled region propositions (including Faster R-CNN), it does not use a delegated region proposal network to speeds up the process. The core of the SSD approach consists of predicting category scores and box offsets for a fixed set of default bounding boxes using small convolutional filters applied to the feature maps. During training, it is necessary to establish the correspondence between the truth of the terrain and the default boxes. Note that for each box, selecting default boxes that vary according to location, aspect ratio and scale. The first step is to match each box to the default box with the best overlap. This is the matching approach used by the original MultiBox and ensures that each box has exactly one matching default box. Then match the default boxes to any fundamental truth with an overlap greater than a threshold (0.5). Adding these mappings simplifies the learning problem: it allows the network to predict high confidences for several overlapping default boxes rather than asking it to select only the one with maximum overlap.

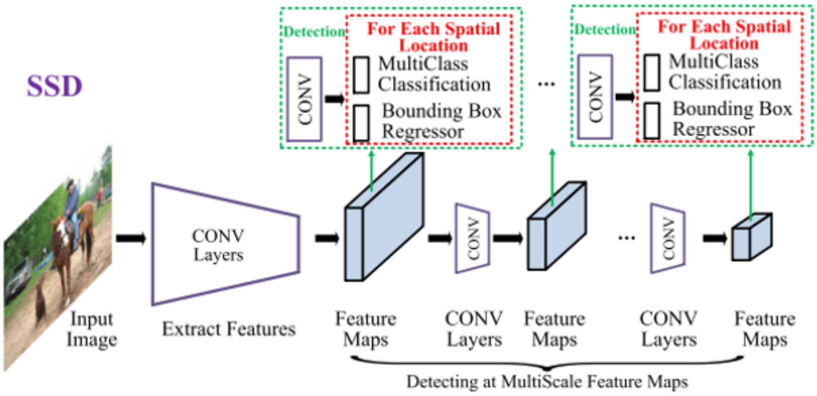


Fig5. SSD workflow

3.3 Comparison

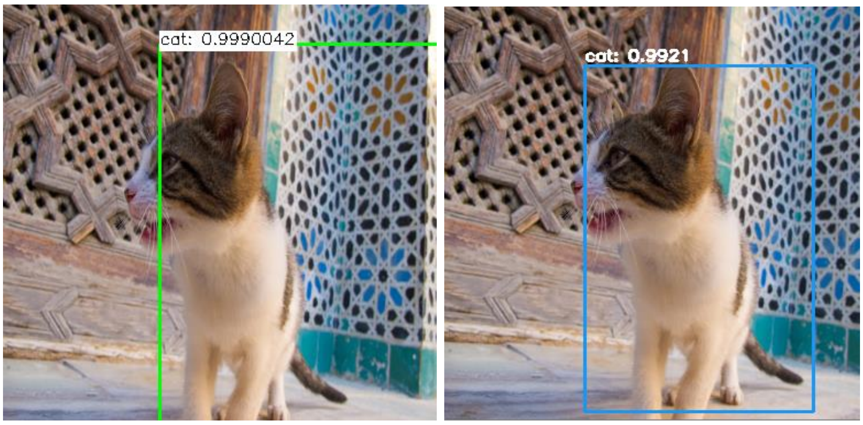
Single-stage detectors use model architectures that directly predict object bounding boxes for a single-step image. In other words, there is no intermediate task (as with two-stage detectors with region proposals) that must be performed to produce a result. This leads to a simpler and faster model architecture. The following table summarizes the performance of each detector [26]:

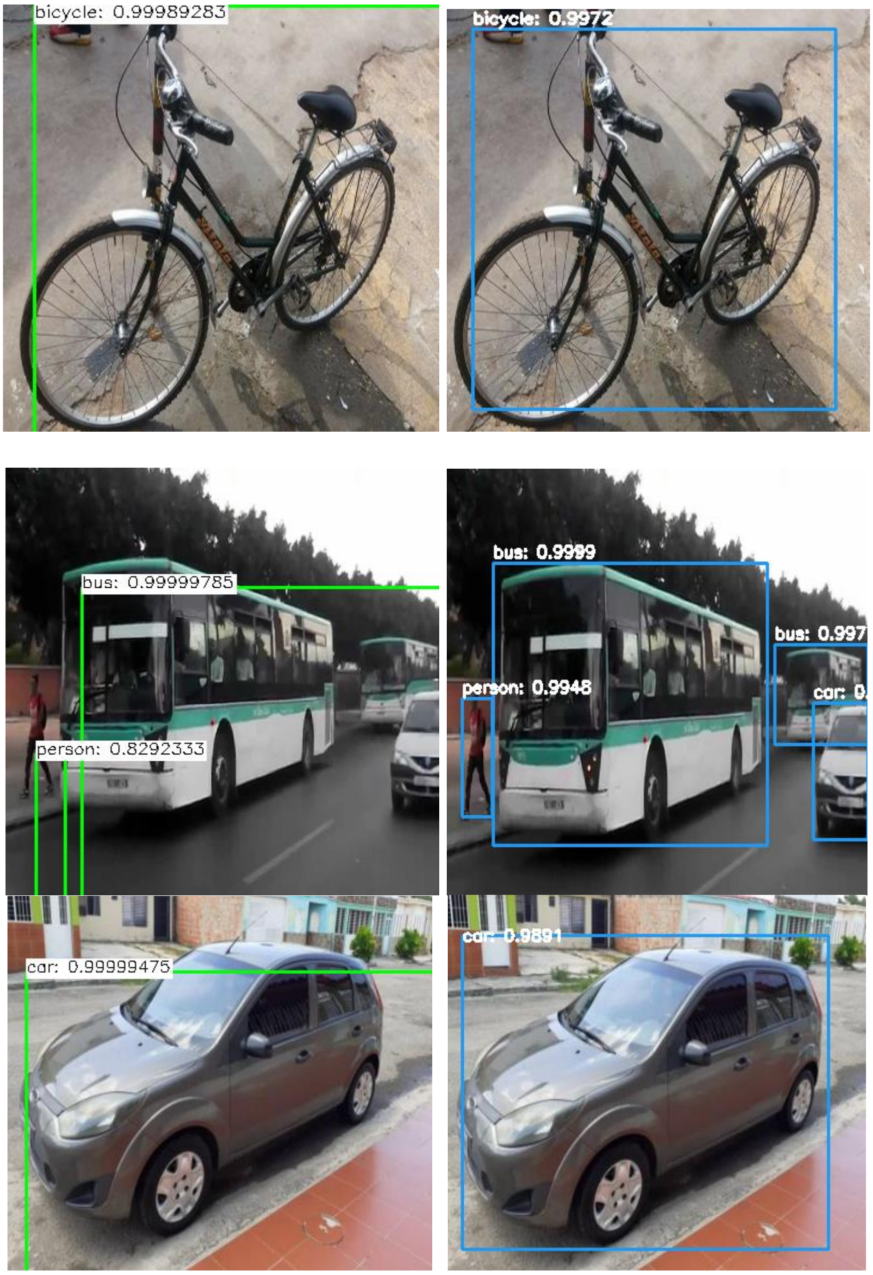
Table 2. Comparison of SSD, Yolo, Yolov2, Yolov3 (Tested on Pascal Voc2012 Dataset)

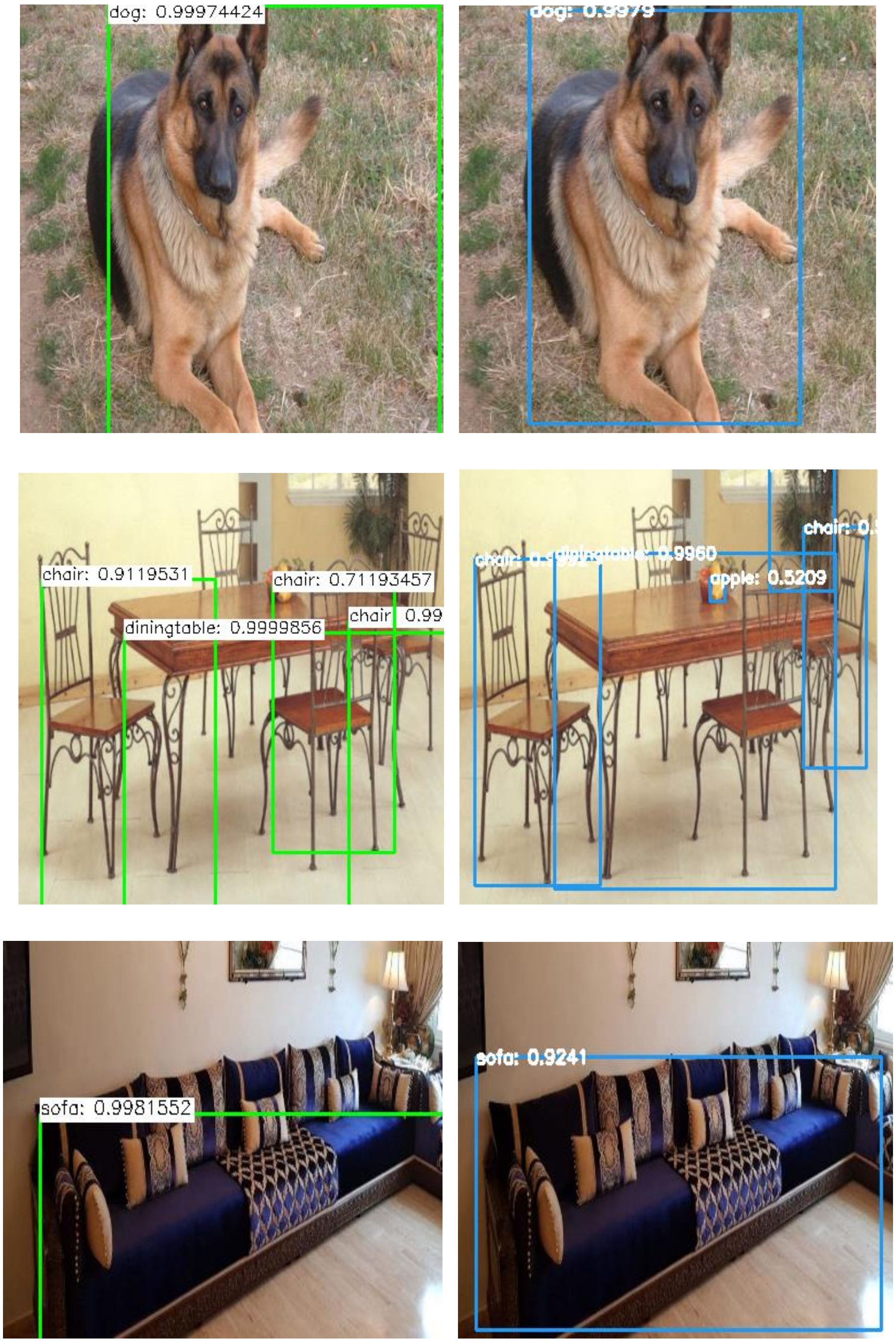
	Yolo	**Yolo v2**	**Yolo v3**	**SSD**
Frame per seconde	45	40	55	46
mAP(%)	63.4	78.6	76.7	74.3

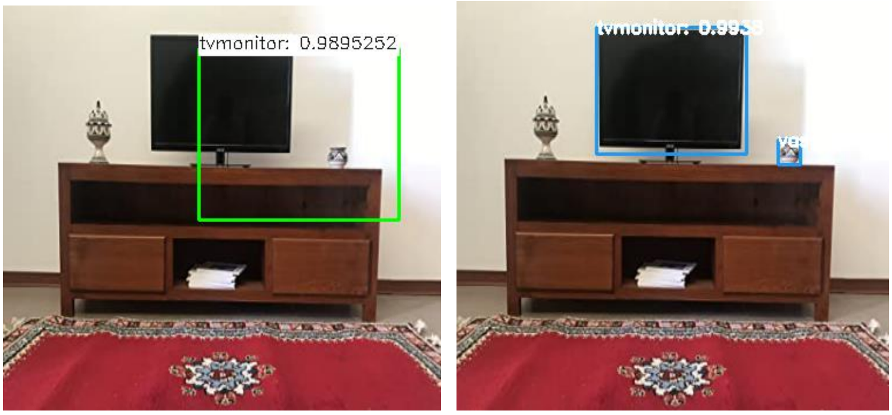
4 Experimentation

To test the performance of the single-stage detectors in real time, we performed tests on different scenes to detect and locate one or more objects. We applied detection with SSD and Yolov3 on the same scene for each test. The real-time object detection application is realized by the OpenCV library using Python as programming language. The scene on the left represents the detection with SSD and the scene on the right represents the detection with Yolo v3.




bicycle: 0.99989283
bicycle: 0.9972
bus: 0.99999785
person: 0.8292333
bus: 0.9999
bus: 0.997
person: 0.9948
car: 0.
car: 0.99999475
car: 0.9891


dog: 0.99974424
dog: 0.9979
chair: 0.9119531
chair: 0.71193457
diningtable: 0.9999856
chair 0.99
chair: 0.
diningtable: 0.9960
apple: 0.5209
chair: 0.
sofa: 0.9981552
sofa: 0.9241



SSD

Yolo v3

Fig6. Applying SDD (left) and Yolo (right) in real-time scenes

Different tests show that Yolo v3 provides slightly lower detection performance than SSD, but it is better for object location. The results were also evaluated in terms of object size and detection accuracy, which showed that large and medium-sized objects were detected with better accuracy with Yolo v3. The following table summarizes the detection performance of these two algorithms in terms of the percentage of the detected class.

Table 3. Comparison of SSD, Yolov3 (detected class precision)

Precision	**Yolo v3**	**SSD**
bike	0.999	0.992
bus	0.999	0.999
car	0.999	0.989
cat	0.999	0.992
chair	0.911	0.999
person	0.824	0.994
table	0.999	0.996
dog	0.999	0.997
sofa	0.998	0.924
tv	0.989	0.993

Since each detection is performed in real time, it was interesting to test the execution time for each test performed. The following table summarizes the performance of the detection of these two algorithms in terms of execution time.

Table 4. Comparison of SSD, Yolov3 (Execution time)

Time(seconde)	Yolo v3	SSD
bike	0.046858	0.015586
bus	0.045902	0.015609
car	0.045212	0.015516
cat	0.051860	0.001024
chair	0.062484	0.015498
person	0.046862	0.015619
table	0.060985	0.015594
dog	0.046822	0.000997
sofa	0.045877	0.000975
tv	0.032944	0.000996

The results show that SSD is twice faster than Yolo v3, we deduce that SSD offers a balanced compromise between precision and speed compared to Yolo v3 which is more performer on the object location.

5 Discussion

YOLO and SSD systems are object detection methods that can quickly recognize objects in images by running a convolutional network on the input image, process the current image in a single step, and calculate a feature map (feature extraction). YOLO v3 is a fast detector, good for real-time processing. Predictions (locations and object classes) are made from a single network. Can be formed end-to-end to improve accuracy. The Single Shot MultiBox Detector (SSD) has achieved good results in object detection, compromising a very modest location. Yolo v3 is a better recommendation for real-time detection, however, if the accuracy of the location is not too important, SSD will be a good choice. A visual reflection of speed versus a compromise of accuracy differentiates them well.

6. Conclusion

Object detection is a crucial step in augmented reality to provide additional information about a certain object in a real scene. The rapid evolution of technology and the exponential increase in the volume of data have given rise to new algorithms and new architectures. The appearance of single-stage detectors have greatly facilitated real-time detection, they aim to speed up detection by removing the region generation step present in two-stage detectors. They attempt to meet the requirements of augmented reality where information flows rapidly. Thus, applications and services of this technology can be enhanced using these algorithms, such as SSD and YOLO. We examined how the integration of these real-time algorithms can improve the quality of experience and quality of service of augmented reality applications. Future work will focus on the use of these algorithms in augmented reality to improve the user experience. More specifically, this work will focus on object recognition under different conditions, the retrieval of relevant information through the exploitation of data and the evaluation of this information in the natural environment of the users in real time and in an interactive way.

References

[1] Z. Oufqir, A. E. Abderrahmani, et K. Satori, « Important Method for Detecting and Tracking Based on Color », vol. 8, n^{o} 5, p. 5, 2019.

[2] Z. Oufqir, A. E. Abderrahmani, et K. Satori, « Comparative Study of Object Insertion Methods on Image Sequence for Augmented Reality », p. 5, 2018.

[3] Y. Zhao, H. Shi, X. Chen, X. Li, et C. Wang, « An overview of object detection and tracking », in *2015 IEEE International Conference on Information and Automation*, août 2015, p. 280- 286. doi: 10.1109/ICInfA.2015.7279299.

[4] X. Du, Y. Cai, S. Wang, et L. Zhang, « Overview of deep learning », in *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, nov. 2016, p. 159- 164. doi: 10.1109/YAC.2016.7804882.

[5] J. Schmidhuber, « Deep learning in neural networks: An overview », *Neural Networks*, vol. 61, p. 85- 117, janv. 2015, doi: 10.1016/j.neunet.2014.09.003.

[6] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, et A. Torralba, « Object Detectors Emerge in Deep Scene CNNs », *arXiv:1412.6856 [cs]*, avr. 2015, Consulté le: juin 18, 2020. [En ligne]. Disponible sur: http://arxiv.org/abs/1412.6856

[7] V. B. Weigel, *Deep Learning for a Digital Age: Technology's Untapped Potential To Enrich Higher Education*. Jossey-Bass, 989 Market Street, San Francisco, CA 94103-1741 ($28), 2002.

[8] J. Wang, Y. Ma, L. Zhang, R. X. Gao, et D. Wu, « Deep learning for smart manufacturing: Methods and applications », *Journal of Manufacturing Systems*, vol. 48, p. 144- 156, juill. 2018, doi: 10.1016/j.jmsy.2018.01.003.

[9] G. Hinton, « Deep Learning—A Technology With the Potential to Transform Health Care », *JAMA*, vol. 320, n^{o} 11, p. 1101- 1102, sept. 2018, doi: 10.1001/jama.2018.11100.

[10] S. Gould, P. Baumstarck, M. Quigley, A. Y. Ng, et D. Koller, « Integrating Visual and Range Data for Robotic Object Detection », p. 13.

[11] C. del-Blanco, F. Jaureguizar, et N. Garcia, « An efficient multiple object detection and tracking framework for automatic counting and video surveillance applications », *IEEE Trans. Consumer Electron.*, vol. 58, n^{o} 3, p. 857- 862, août 2012, doi: 10.1109/TCE.2012.6311328.

[12] J. K. Schiffmann et N. Park, « METHOD FORESTIMATING UNKNOWN PARAMETERS FORAVEHICLE OBJECT DETECTION SYSTEM », p. 12.

[13] R. H. Baer et J. B. Grand, « Object detection for an interactive human interface device », US20080039199A1, févr. 14, 2008 [En ligne]. Disponible sur: https://patents.google.com/patent/US20080039199/en

[14] « Image Background Search: Combining Object Detection Techniques with Content-Based Image Retrieval (CBIR) Systems », *ResearchGate*. https://www.researchgate.net/publication/2375669_Image_Background_Search_Combining_Object_Detection_Techniques_with_Content-Based_Image_Retrieval_CBIR_Systems

[15] K. A. Joshi et D. G. Thakore, « A Survey on Moving Object Detection and Tracking in Video Surveillance System », vol. 2, n^{o} 3, p. 5, 2012.

[16] Z. Oufqir, A. E. Abderrahmani, et K. Satori, « Inserting and tracking a plane object in a three- dimensional scene. », p. 14.

[17] Z. Oufqir, A. El Abderrahmani, et K. Satori, « From Marker to Markerless in Augmented Reality », in *Embedded Systems and Artificial Intelligence*, Singapore, 2020, p. 599‑ 612. doi: 10.1007/978-981-15-0947-6_57.

[18] R. Girshick, J. Donahue, T. Darrell, et J. Malik, « Rich feature hierarchies for accurate object detection and semantic segmentation », *arXiv:1311.2524 [cs]*, oct. 2014, [En ligne]. Disponible sur: http://arxiv.org/abs/1311.2524

[19] R. Girshick, « Fast R-CNN », *arXiv:1504.08083 [cs]*, sept. 2015, [En ligne]. Disponible sur: http://arxiv.org/abs/1504.08083

[20] S. Ren, K. He, R. Girshick, et J. Sun, « Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks », *arXiv:1506.01497 [cs]*, juin 2015, [En ligne]. Disponible sur: http://arxiv.org/abs/1506.01497

[21] J. Redmon, S. Divvala, R. Girshick, et A. Farhadi, « You Only Look Once: Unified, Real-Time Object Detection », *arXiv:1506.02640 [cs]*, mai 2016, [En ligne]. Disponible sur: http://arxiv.org/abs/1506.02640

[22] W. Liu *et al.*, « SSD: Single Shot MultiBox Detector », *arXiv:1512.02325 [cs]*, vol. 9905, p. 21‑ 37, 2016, doi: 10.1007/978-3-319-46448-0_2.

[23] K. He, X. Zhang, S. Ren, et J. Sun, « Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition », *arXiv:1406.4729 [cs]*, vol. 8691, p. 346‑ 361, 2014, doi: 10.1007/978-3-319-10578-9_23.

[24] J. Redmon et A. Farhadi, « YOLO9000: Better, Faster, Stronger », *arXiv:1612.08242 [cs]*, déc. 2016, [En ligne]. Disponible sur: http://arxiv.org/abs/1612.08242

[25] J. Redmon et A. Farhadi, « YOLOv3: An Incremental Improvement », *arXiv:1804.02767 [cs]*, avr. 2018, [En ligne]. Disponible sur: http://arxiv.org/abs/1804.02767

[26] Z.-Q. Zhao, P. Zheng, S. Xu, et X. Wu, « Object Detection with Deep Learning: A Review », *arXiv:1807.05511 [cs]*, avr. 2019, Consulté le: mai 26, 2020. [En ligne]. Disponible sur: http://arxiv.org/abs/1807.05511

Notes on contributors

Zainab OUFQIR PhD student at university sidi mohamed ben abdellah, LISAC Department of Mathematics and informatic Faculty of Sciences Dhar-Mahraz P.O.Box 1796 Atlas Fes, 30000, Morocco. Email: zainab.oufkir@usmba.ac.ma

Abdellatif EL ABDERRAHMANI ¨Professor at university sidi mohamed ben abdellah, LISAC Department of Mathematics and informatic Faculty of Sciences Dhar-Mahraz P.O.Box 1796 Atlas Fes, 30000, Morocco.

Email: abdellatif.elabderrahmani@usmba.ac.ma

Khalid SATORI ¨Professor at university sidi mohamed ben abdellah, LISAC Department of Mathematics and informatic Faculty of Sciences Dhar-Mahraz P.O.Box 1796 Atlas Fes, 30000, Morocco. Email: khalid.satori@usmba.ac.ma