Accurate object detection system on HoloLens using YOLO algorithm

Haythem Bahri*, David Krčmařík*, Jan Kočí*
*Institute for Nanomaterials, Advanced Technologies and Innovation
Technical University of Liberec
Liberec, Czech Republic
haythem.bahri@tul.cz, david.krcmarik@tul.cz, jan.koci@tul.cz

***Abstract*—We demonstrate in our paper, an implementation on Microsoft HoloLens, deep learning supported in the context of object detection. The main aim of this system is to create the more accurate object detection model for Augmented Reality using communication between the deep learning processing and the Microsoft HoloLens as Input/Output device. This system aims to help the wearable device user to detect and to recognize between objects in real world. For the object detection approach, a deep learning model has been used for the implementation of this system called YOLO. This model is near to real-time and it supports to detect more than 9000 objects. Our system provides the annotation of augmented object detected and its limitation area or bounding box via HoloLens. It allows to detect the new position of moving object in a few milliseconds. Preliminary results show a great rate of object detection with a detection time comparable.**

***Keywords*-Augmented Reality; Microsoft HoloLens; Object detection; You Only Look Once; detection time; deep learning**

I. Introduction

Object detection technology is to extract effective detection information, identify and recognize by computer instead of natural person. In addition, it is difficult to forge, cannot be lost, portable, and easy to use. It overcomes the shortcomings of traditional identity authentication methods and provides a more secure and reliable authentication mechanism. Therefore, as the basic application of artificial intelligence technology, object recognition has been widely used in public safety and enterprise management. This has reduced the cost of the industry to a certain extent, improved service efficiency and management level, and has been widely recognized by people from all walks of life.

During the development of human-computer interaction, it is more and more easy for users to accept the more practical, more human, and more intelligent interaction mode. The application of human-computer interaction technology in the field of graphics and images is to pursue a better user experience. Object detection technology can reduce the number of user hands-on operations and provide a new way of interaction that is different from finger touching the screen, truly achieve the liberation of hands, to achieve more convenient, less burden of human-computer interaction. However, the mainstream interactive devices related to object detection are still traditional keyboard, mouse, and touch screen. However, these devices have many limitations. The utilities to detect object using video-surveillance systems is already proved its efficiency regarding the revolution of deep learning algorithms.

Otherwise, these systems require more than device to realize the object detection system. First device stock the scene of the objects and the second apply the deep learning algorithm to provide the result of detection and to recognize the object. From Nowadays and due to Microsoft device HoloLens, it could be visualizing, detecting and recognizing the overall object in front of you in real world using this smart device which is easier than earlier. This can improve everyday life quality and help with environmental orientation and human-computer interaction. This prototype can be used in several field as Building Information Modeling, military attacks, auto design ... etc.

In this work, we developed an object detection system to detect and to recognize the object via Microsoft HoloLens at user and applying YOLO or You Only Look Once as a deep learning algorithm at server side to process the data from the user side or the client side. In order to let the HoloLens communicate with the server, a custom protocol over TCP/IP is employed. This communication works through a local network and it is able to transfer both streaming video and data deriving from the recognition.

Our paper is divided as follows, section 2 covers a related work in the field of object detection using HoloLens with a study about the deep learning algorithms, as case of this study: YOLO. In section 3, we describe: our object detect system, the implementation on HoloLens and the utility of YOLO algorithm in this approach. In section 4 a preliminary evaluation of the results of detection and discussion. The last section 5 present a conclusion of this contribution providing ideas for future possibilities.

II. Related work

The facilities to implement the application on the Microsoft HoloLens device makes the users run to bring distances closer to the latest algorithms of deep learning for object detection as CNN, RCNN, fast-RCNN, faster-RCNN and YOLO. In this section, we present an overview of the researches that use the YOLO algorithms for object detection. In the same way, we show the researches that use HoloLens for object detection approach.

978-1-7281-3572-4/19/$31.00 ©2019 IEEE
DOI 10.1109/ICCAIRO47923.2019.00042

A. Object detection algorithms: case of study "YOLO"

The approach presented by Redmon in 2016 [1] has introduced YOLO as a unified model for object detection. His model is as simple to construct as it can be trained directly on full images. Unlike to other approaches that use classifiers, Redmon's algorithm is trained on a loss function that directly corresponds to detection performance and the entire model is trained jointly. In this contribution, he introduced the fastest approach algorithm for object detection in the literature when he contributes to push the state of the art in real-time application. YOLO approach defined an ideal compromise that rely on fast and robust object detection.

A new approach was developed by Redmon in 2017, we talk about YOLOv2 and YOLO9000, the second versions of YOLO algorithm for real-time detection systems. YOLOv2 is considered as the faster than other detection systems across a variety of detection datasets as Redmon shown in [2]. Therefore, this approach demonstrates a best result when it run at variety of images size to provide a smooth trade-off between speed and accuracy. In other hand, YOLO9000 provides a real-time algorithm to detect more than 9000 objects that optimize simultaneously detection and classification. The author used Word Tree model to include data from different dataset by joining the optimization technique to train ImageNet and COCO in the same time. This model offers more detailed output space for image classification, when the combination of datasets using a hierarchical classification was used in the classification and segmentation domains. He used a multi-scale technique for the training to provide a benefit across a variety of visual task. This version of YOLO made a strong step to close the dataset size gap between detection and classification.

The work of Redmon not finished here, the newest version of YOLO or as we can say some updates was made in 2018, it is called YOLOv3. He presented a bunch of little design changes to make it better, when it is pretty swell and little bigger but more accurate. YOLOv3 still fast, this was presented in his work in [4] for different size of images. Many works have used these algorithms of YOLO for the object detection. As we can notice the work of Benjdira [3], when he presented a comparison of an experimental results of the implementation between Faster RCNN and YOLOv3 for cars detection task from UAV images. For the performance evaluation, he adopted in five metrics: precision, recall, F1 score, quality and processing time. The results of both algorithms are comparable in precision. However, YOLOv3 outperforms Faster RCNN in sensitivity which is more capable to extract all cars in image with 99.07% accuracy. In the same way, YOLOv3 outperforms Faster RCNN in the processing time for one image detection.

Other research has used YOLOv3 for cyclist detection in large high-resolution images in the work of Liu [5]. The author used a YOLOv3 network, ACF-PR region proposal method, and a post-processing step in order to extract potential regions from high resolution images. Firstly, Liu used the ACF detector to fast extract candidates then a bounding box merging and extending method is designed to merge the bounding boxes into correct region proposals to be prepared for the YOLO algorithm. After that, the authors applied the YOLOv3 algorithm to design the detection in the potential regions generated by ACF-PR. A comparative study was made between his method and other representative methods using the pubic TDCB dataset and the experimental results shown that his method outperforms YOLOv3 by 13.69% average precision and outperforms SSD (Single Shot MultiBox Detector) by 25.27% average precision.

Derakhshani in [8] has shown an improvement version of YOLO algorithms in terms of mAP (mean average precision) without compromising their speed. In the training step of YOLO algorithms, he feed the network learn with some localization information as a post-processing to help the network (YOLO algorithms) to better localize. Of course, he gradually reduces his assisted excitation to zero during the latest stages of training. His technique improves the mAP of YOLOv2 by 3.8% an YOLOv3 by 2.2.% on MSCOCO dataset.

B. Object detection using HoloLens

For an efficient and accurate manufacturing assembly fault detection, Wang has used in [6] the HoloLens mixed reality to be incorporated as an input/output module and the Faster RCNN for targeting recognition to obtain mass data extraction feature information replacing manual inspection. In the training stages, he used 2000 samples of objects, the target detection reaches an 85% as mAP.

The work of Eckert in [7] shown that recent technology advances in deep learning and mixed reality hardware allow faster development of assistive technologies. He proposed a system that offers possibilities to simplify the everyday life of those who are visually impaired or blind and can be used without any previous training on the HoloLens to find the basic objects. His prototype aims to substitute the impaired eye of the user and replace it with technical sensors by scanning its surroundings. The prototype provides a situational overview of objects around the HoloLens by the implementation of YOLOv2. This prototype can display and read out the name of augmented objects which can be selected by voice commands and used as directional guides for the user, using 3D audio feedback. A distance announcement of a selected object is derived from the HoloLens's spatial model. The wearable solution offers the opportunity to efficiently locate objects to support orientation without extensive training of the user.

In other hand, Corneli and Naticchia in [9] [10] have used the YOLOv2 on HoloLens for BIM/AIM or Building and Asset Information Models. Their research aims to provide

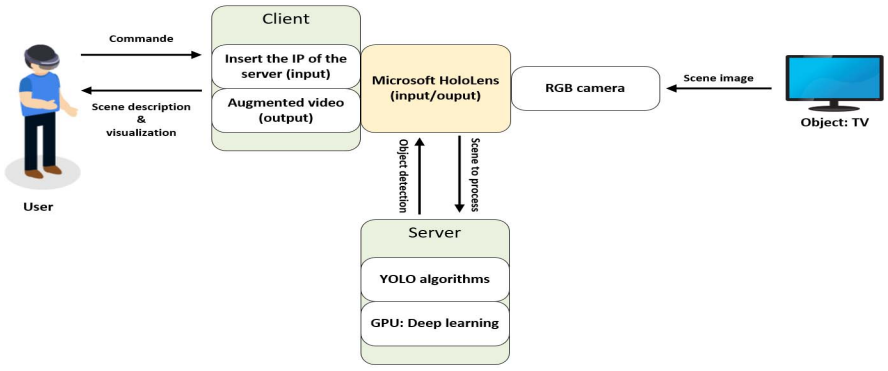


Figure 1. Diagram of object detection system : description of each component and its process

a support to Facility Management operations during the building life cycle. Their system combined three different technologies: the AR, the BIM data model and deep learning, all in one embedded solution for on-site use. The both cases described in their work improved their efficiency thanks to the effective human-computer interaction. The feasibility tests have been conducted related to the training of Neural Network or YOLOv2 and the recognition through the AR device.

III. Object detection system via HoloLens

In this section, we describe the main process of our system to ensure the object detection using YOLO algorithms via HoloLens. The software development contribution is based on client-server interaction connection as we shown in figure 1 above. For the hardware material, we used two Graphics processing Units (GPU) of NVIDIA type Quadro P4000 to perform the computational time of the server side usually the processing of YOLO algorithms. A Microsoft HoloLens version 1 was used for the client side to benefit of its camera as an input scene to object detection in real world. This camera mounted on the front of the device which enables applications to see what the user sees. Developers have access to and control of the camera just as they would for color cameras on smartphones, portables, or desktops. In our system, we imported HoloToolkit the library of Microsoft to control gaze, gesture and cameras. For the server side, we included the darknet library of Redmon [11] to run the YOLO algorithms which is implemented in C and CUDA programming language. All these details are summarized in the table. 1 in the follow.

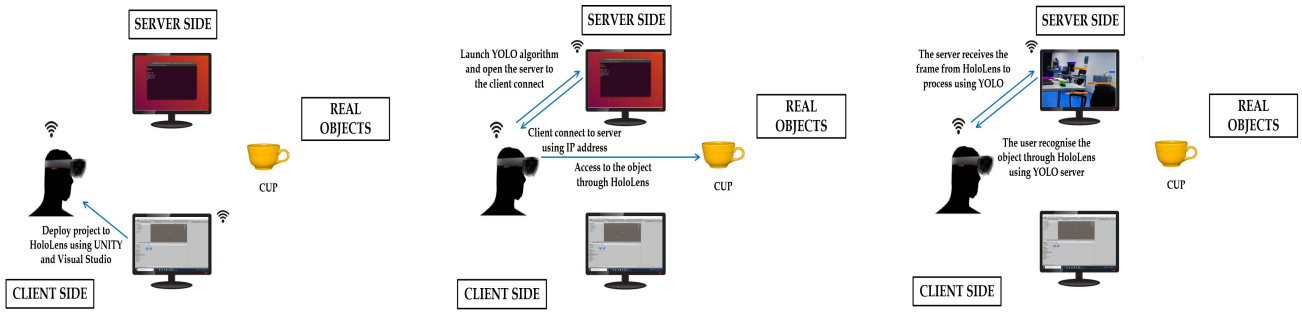
Table I
Development tools from the server to the client

Development stack	
Visual studio	Microsoft VS2017
Unity	v. 2018.3.7f1
HoloToolkit	v. 2017.4.3.0
Server stack	
Ubuntu OS	v. 18.04
Darknet	YOLO Library
YOLOv1, YOLOv2, YOLOv3	Deep learning algorithms
GPU	2 x Quadro P4000
CUDA	NVIDIA Toolkit v10.0
CUDNN	NVIDIA v7.3
OpenCV	Computer Vision library v3.4.0
Client stack	
HoloLens	Microsoft Device v.1

A. Server-side

In the server-side, where we applied YOLO the heaviest part of our system to detect and to recognize the object. We used the darknet library of Redmon [11] by adding the processing file of the server to receive the data from the client and to make it as input data of YOLO algorithm .First, the server requests the input data from the client side which is the camera of our HoloLens to be in real-time detection as a regular Red Green Blue (RGB) frame with the resolution of 896×504. As we shown in figure 2, the first step and after to deploy the application of the client on HoloLens, it would be able to connect using the IP address of the server. In this case, the server receives the frame to perform the object detection. Then, the users launch the considerable network of YOLO to process the frame received. A different pre-trained model of YOLO were used to provide the results to the client. These results are the annotation, the bounding box and the color of the box of each object detected. Beside these results, the probability that an object belongs to a certain class inside the bounding box is calculated and shown with the annotation. For each frame, the server send these results to client to apply it properly on HoloLens with efficiency.

When the client received these results, the user can see the object detection by the HoloLens in real-time with all details. The server is returning 2D coordinates to the HoloLens, then 3D real-world coordinates of the object are calculated on the device based on the received 2D data and the virtual 3D model, generated by the HoloLens. The virtual 3D model is



(a) Deploy the application to client

(b) Connection to server and provide the frames

(c) Object detection and real visualization via HoloLens

Figure 2. The concept of our system from the client to the server to object detection on HoloLens using YOLO algorithm

mapped by a real-time process, supported by the HoloLens Time-of-Flight (TOF) system and visible light cameras. This is necessary to provide distance announcements for the individual recognized objects. The estimation of real world coordinates is achieved by using the spatial model of the Holo-Toolkit. While moving, the TOF and RGB cameras are mapping the surroundings and create a spatial model of the real environment. The virtual scene is generated by scanning and mapping the environment while HoloLens is active. Client-server communication is implemented via TCP/IP Internet connection. This communication has been implemented in C programming language which also ensures that multiple clients can communicate with the server.

B. Client-side

In the side of the users, the objects that will be detected and recognized must inserted via a camera of HoloLens. In this stage, we created a project using the software Unity to be built and deployed on HoloLens using the component of the HoloToolkit. In this project, we started to upload the main camera from HoloToolkit, then we applied the Spatial Mapping step to realize the mapping of the real environment front the users. Before to access to the camera, we need a method to connect the client to the server using TCP/IP interface. To ensure this connection, we used the interface of keyboard from HoloToolkit to insert the IP address of the server. We applied on this keyboard the gaze and the gesture manager from the cursor component. After that, we defined the annotations option and we linked it to the main camera to ensure the show of the object annotation. In figure 2, we presented an overview of the main steps of our system front the client.

IV. Results and Discussion

Our approach to detect and to recognize object through HoloLens show an excellent result of detection. We used the three versions of YOLO to extract its results in term of mean average of precision and processing time. To be efficient in detection application, we determined the number of frames per second for each model of YOLO.

Figure 3. Object detection using HoloLens with YOLO algorithm in the screen

In figure 3, we show a demonstration of object detection using YOLO via HoloLens with the header of each object containing its name and percentage of detection. These figures have extracted from the screen during launching the application using the input from HoloLens and the output in the screen. In figure 4, we present the object detection using the input and the output frame from HoloLens. The both results were in the perfect conformance due that the results of detection, the position of objects and the percentage of detection sent to HoloLens in the same time of shown in screen.

Table II
AVERAGE PRECISION OF OBJECTS DETECTION FOR A VARIETY METHODS OF YOLO

Model	TV monitor	Chair	Keyboard	Mouse	Bottle	Person	Cell phone
YOLOv1	76.07	50.12	79.09	78.61	63.14	86.52	66.56
YOLOv2	83.45	65.87	89.3	87.11	66.72	87.55	78.2
YOLOv3	95.55	98.41	91.93	99	97.83	99	92.23

Figure 4. Object detection using YOLO through HoloLens

A. Precision of detection

The performance of our approach is to detect and recognize the object with a best precision and with a best time. For the precision, we used the 3 versions of YOLO to prove which one is the most efficient in precision. Our results in table II show the efficiency of YOLO algorithms for the average precision of different object in our office. However, YOLOv3 presented a great precision which is more than 90% for overall object detected. In the same way, to confirm these results, we have to provide the mean average precision mAP of these YOLO versions. As we show in table III, the mAP of YOLOv3 is around to 96%. However, YOLOv1 and YOLOv2 can't reach 80%.

Table III
MEAN AVERAGE PRECISION OF DIFFERENT MODEL OF YOLO

Measure	*YOLOv1*	*YOLOv2*	*YOLOv3*
mAP	71.44	79.74	96.27

B. Processing times

We validated our work with computing the processing time of our approach. In table IV, we presented the mean processing time in term of number of frames per second of the 3 models used of YOLO. YOLOv3 has approved its performance in precision of detection and now in term of fast than the first versions of YOLO. It reaches 5 fps to detect all object with a very high precision through HoloLens.

Table IV
THE MEAN PROCESSING TIME OF EACH MODEL OF YOLO DURING OBJECT DETECTION

Model	*YOLOv1*	*YOLOv2*	*YOLOv3*
FPS	4.2	4.6	5

V. CONCLUSION AND FUTURE WORK

In this paper, we shown that the recent technology as augmented reality device HoloLens could be used as input and output of a deep learning applications as object detection. In our work, we created an algorithm between HoloLens as a client and a desktop as a server using TCP/IP internet connection. The aim of this algorithm is to process the object detection and recognition of the augmented reality from the HoloLens using a server based on YOLO algorithm. Our work presents an improvement results of detection via HoloLens as 96% mAP for YOLOv3. For the processing time, the detection and the recognition were processed in 5 fps for YOLOv3.

As a future work, the second generation HoloLens includes the AI coprocessor which built into its HPU for its central vision-processing chip. This AI coprocessor is used to analyze the data of deep neural networks, one of the principal tools of contemporary AI. The approach of HoloLens 2 can make this application more efficient, easy and more speed to reach the real-time detection of object.

ACKNOWLEDGMENT

The result was obtained through the financial support of the Ministry of Education, Youth and Sports of the Czech Republic and the European Union (European Structural and Investment Funds - Operational Programme Research, Development and Education) in the frames of the project "Modular platform for autonomous chassis of specialized electric vehicles for freight and equipment transportation", Reg. No. CZ.02.1.01/0.0/0.0/16 025/0007293.

REFERENCES

[1] Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. "You only look once: Unified, real-time object detection." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779-788. 2016.

[2] Redmon, Joseph, and Ali Farhadi. "YOLO9000: better, faster, stronger." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7263-7271. 2017.

[3] Benjdira, Bilel, Taha Khursheed, Anis Koubaa, Adel Ammar, and Kais Ouni. "Car detection using unmanned aerial vehicles: Comparison between faster r-cnn and yolov3." In 2019 1st International Conference on Unmanned Vehicle Systems-Oman (UVS), pp. 1-6. IEEE. 2019.

[4] Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." arXiv preprint arXiv:1804.02767. 2018.

[5] Liu, Chunsheng, Yu Guo, Shuang Li, and Faliang Chang. "ACF Based Region Proposal Extraction for YOLOv3 Network Towards High-Performance Cyclist Detection in High Resolution Images." Sensors 19, no. 12: 2671.2019.

[6] Wang, Shuai, Ruifeng Guo, Hongliang Wang, Yuanjing Ma, and Zixiao Zong. "Manufacture Assembly Fault Detection Method based on Deep Learning and Mixed Reality." In 2018 IEEE International Conference on Information and Automation (ICIA), pp. 808-813. IEEE. 2018.

[7] Eckert, Martin, Matthias Blex, and Christoph M. Friedrich. "Object detection featuring 3D audio localization for Microsoft HoloLens." In Proc. 11th Int. Joint Conf. on Biomedical Engineering Systems and Technologies, vol. 5, pp. 555-561. 2018.

[8] Derakhshani, Mohammad Mahdi, Saeed Masoudnia, Amir Hossein Shaker, Omid Mersa, Mohammad Amin Sadeghi, Mohammad Rastegari, and Babak N. Araabi. "Assisted Excitation of Activations: A Learning Technique to Improve Object Detectors." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9201-9210. 2019.

[9] Naticchia, B., A. Corneli, A. Carbonari, A. Bonci, and M. Pirani. "Mixed reality approach for the management of building maintenance and operation." In ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction, vol. 35, pp. 1-8. IAARC Publications. 2018.

[10] Corneli, A., B. Naticchia, A. Carbonari, and F. Bosché. "Augmented Reality and Deep Learning towards the Management of Secondary Building Assets." In ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction, vol. 36, pp. 332-339. IAARC Publications. 2019.

[11] Redmon, J. YOLOv3. https://pjreddie.com/darknet/yolo/ [Accessed: January 15, 2020].