

## Alzheimer's diseases classification using YOLOv2 object detection technique

Sara Saad Abd-Aljabar<sup>1</sup>, Nasseer Moyasser Basheer<sup>2</sup>, Omar Ibrahim Alsaiif<sup>3</sup>

<sup>1</sup>Department of Computer Engineering Technology, Northern Technical University, Mosul, Iraq

<sup>2</sup>Department of Medical Instrumentation Techniques Engineering, Northern Technical University, Mosul, Iraq

<sup>3</sup>Department of Computer Systems Technologies, Northern Technical University, Mosul, Iraq

### Article Info

#### Article history:

Received Jan 22, 2022

Revised Apr 15, 2022

Accepted Jun 11, 2022

#### Keywords:

Alzheimer's disease

Deep learning

Magnetic resonance imaging

Object detection

YOLOv2

### ABSTRACT

Early diagnosis and treatment of Alzheimer's disease (AD) is necessary for the patient safety. Computer-aided diagnosis (CAD) is a useful tool for early diagnosis of Alzheimer's disease (AD). We make two contributions to the solution of this problem in this study. To begin with, we are the first to propose an Alzheimer's disease diagnosis solution based on the MATLAB that does not require any magnetic resonance imaging (MRI) pre-processing. Second, we apply recent deep learning object detection architectures like YOLOv2 to the diagnosis of Alzheimer's disease. A new reference data set containing 300 raw data points for Alzheimer's disease detection/normal control and severe stage (MCI/AD/NC) deep learning is presented. Primary screening cases for each category from the Alzheimer's disease neuroimaging initiative (ADNI) dataset. The T1-weighted digital imaging and communications in medicine (DICOM) MRI slice in the MP-Rage series in 32-bit DICOM image format and 32-bit PNG are included in this dataset. By using MATLAB's image label tool, the test data were marked with their appropriate class label and bounding box. It was possible to achieve a detection accuracy of 0.98 for YOLOv2 in this trial without the usage of any MRI preprocessing technology.

*This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.*



### Corresponding Author:

Sara Saad Abd-Aljabar

Department of Computer Engineering Technology, Northern Technical University

Mosul, Iraq

saraalbadrany@gmail.com

## 1. INTRODUCTION

The most common type of dementia is Alzheimer's disease (AD). In developed countries Alzheimer's disease is estimated to affect almost 5% of those over the age of 65 years and a shocking 30% after the age of 85 years. It is expected that Alzheimer's disease will affect 0.64 billion individuals by 2050 [1]. The syndrome begins with a slight loss of memory and progresses over time. Magnetic resonance imaging (MRI) is a non-radioactive imaging technique that can be used to detect the disease. An MRI can be utilized to determine the disease's form, size, and location. A full medical evaluation is required for psychologists to detect Alzheimer's disease. As well as additional physical and neurological examinations. Manually categorizing and analyzing these images is a challenging task for technicians. As a result, the use of automatic detection systems has become increasingly popular. Using different methods of the neural network, the distortions can be easily detected.

There are seven stages of Alzheimer's disease development: stage 1: no harm-there are no difficulties with remembering or other signs of dementia at this stage, so Alzheimer's disease cannot be detected; stage 2: slight deterioration-adults 65 and older may experience mild memory issues, but not to the

point where dementia is distinguishable from regular Alzheimer's disease; stage 3: moderate deterioration-difficulties with memory and knowledge appear at this stage, and clinicians can detect reduced cognitive ability; stage 4: milder regression-individuals with stage four will experience short-term memory loss as well as difficulties with basic arithmetic; stage 5: Alzheimer's patients with stage five require assistance with daily tasks due to moderately severe impairment; stage 6: severe regression-Alzheimer's patients in stage 6 require close supervision as their ability to carry out basic daily activities deteriorates; and stage 7: severe deterioration-individuals lose their ability to react to their living environment and interact at this stage of the disease [2]. To survive, patients at this stage require around-the-clock care.

The hippocampus is the part of the brain where episodic and spatial memory are stored. It also serves as a connection between our bodies and minds. When the hippocampus shrinks, it loses cells and damages synapses and nerve endings. As a result, connections between neurons no longer work. As a result, areas of the involved in recalling (short-term memory), learning, decision making, and judging are impaired [3]. A general diagnosis of dementia requires a detailed examination that includes medical history taking, analysis of cognitive function and mental status, neurophysiological testing, assessment of activities of daily living, clinical laboratory testing, and brain imaging tests [4], [5].

Early diagnosis of Alzheimer's disease is critical for disease management and preventing loss of abilities to accomplish even basic tasks. Machine learning algorithms use a variety of algorithms to extract, analyze and learn the necessary data from MRI scans and create a prediction based on the input data [6]. As a result, the system is "trained" applying algorithms in order to fulfill the task, enabling it to understand and carry out the diagnosis task. Cascades of different processing units are used to extract and transform information. Each successive layer takes its input from the output of the previous layer. The methods employed in pattern analysis applications may be supervised, whereas classification applications may be unsupervised.

The Alzheimer's disease neuroimaging initiative (ADNI) database was created to assist clinicians and researchers by creating an MRI database based on clinical dementia rating (CDR). ADNI was founded as public-private collaboration in 2003 [7]. Principal investigator Michael W. Weiner, MD, is in charge of the cooperation. The major purpose of the ADNI study was to see if serial MRI, PET, other biological markers, and clinical and neuropsychological assessments could be used to track the evolution of mild cognitive impairment (MCI) and early Alzheimer's disease (AD).

To assess the progression of MCI and the early stages of Alzheimer's disease (AD), a range of neuropsychological tests can be performed. A huge amount of MRI and PET scan data is stored in the ADNI database. Blood and cerebrospinal fluid biomarkers, as well as photos Analyzes of three different types of persons: healthy people controls (NC), Alzheimer's sufferers, and patients with MCI symptoms.

A modern deep learning object recognition components, such as YOLOv2\* was used to diagnose Alzheimer's disease. In this study an Alzheimer's disease diagnosis approach that does not require any form of MRI preprocessing, is suggested. It was possible to present a new deep learning AD/NC object detection sample dataset derived out from ADNI database, which includes 200 raw, unprocessed scan states for each category for future item identification study [8].

## 2. OBJECT DETECTION USING DEEP LEARNING

Object detection in general should be fast, precise, and capable of identifying a wide range of objects. Since the introduction of neural networks, detection frameworks have become more quick and precise. Most detection methods, on the other hand, are still confined to a small number of things. This work proposes a new way to use the large amount of classification data already available and use it to extend the range of existing detection systems, it was also suggested a common object detector train technique based on detection and classification data. This approach uses labeled detection photos to learn reliably locate things, as well as classification images to expand its capacity.

The YOLO end-to-end object recognition algorithm [9] was proposed by Redmond *et al.* In the year 2016, YOLO divides the input image into  $S \times S$  grids and predicts the probability of bounding box B and class C for each grid cell. As seen in Figure 1, each cell has a grid. Each bounding box has five predictions: w, h, x, y, and a confidence object. The values w and h represent the box's width and height in relation to the total image. The (x, y) values indicate the coordinates of the square's center in relation to the grid cell's bounds. The term "object confidence" relates to the item's ability to be trusted. which is defined as (1).

$$Confidence = Pr(object) \times IOU_{pred}^{truth} \quad (1)$$

In (1) PR (object) denotes the probability that the object will fit into the current grid cell. The predicted bounding square and the real square intersect on union (IOU), which is represented as a fact in

front of the IOU. The majority of bounding boxes with low confidence in the item are then discarded under the set limit. Finally, using non-maximum suppression (NMS) [9], unwanted bounding boxes are deleted. The procedure is used to get rid of any bounding boxes that aren't needed.

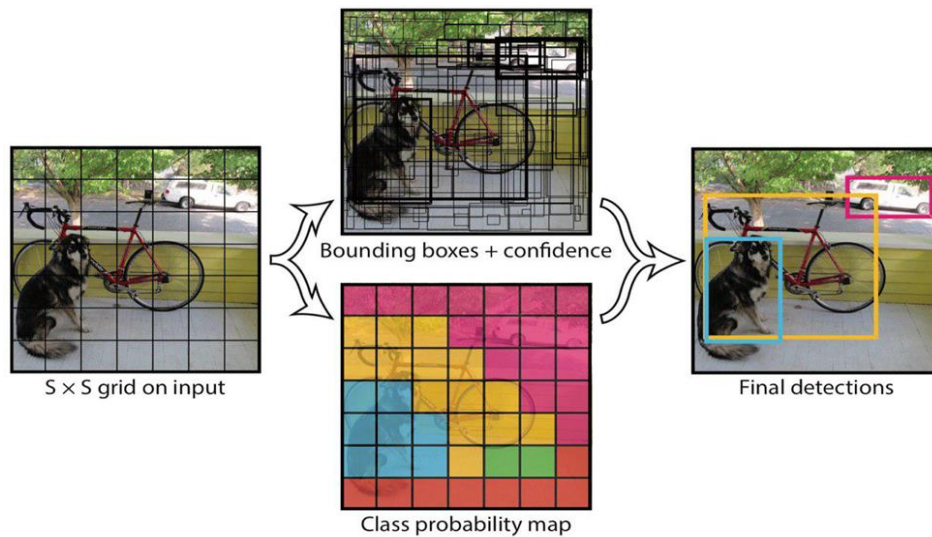


Figure 1. Predicting objects in the image

Methods for generating probable bounding boxes in a picture and then using a classifier to classify them to refine bounding boxes, remove duplicate detections, and reclassify boxes depending on other items in the scene, post-sort processing is used [10]. Since each component must be taught separately, these complex pipelines are slow and difficult to improve. Object recognition is framed as a single regression problem that proceeds directly from image pixels to bounding box coordinates and class probabilities. It is easy to figure out what is present and where it is in the image using this strategy, where you only look once (YOLO). Figure 1 shows how simple YOLO is.

A single convolutional neural network predicts multiple bounding boxes and class probabilities for these boxes at the same time. By training on complete images, YOLO increases detection performance. Compared to traditional object detection approaches, this network has significant advantages.

To begin with, YOLO is really quick. So, it is not required to use a complex pipeline because the detection framework as a regression problem is already defined. When testing, simply run the neural network on a fresh image. It's time to make predictions about what will be discovered. On Titan X, the core network runs at 45 frames per second without batch processing.

The GPU-accelerated version has a frame rate of more than 150 frames per second. This is incredible. It means that the video stream in real time can be processed with a reaction time of less than 25 milliseconds. Furthermore, in real time, YOLO achieves more than twice the average accuracy of competing systems. Using YOLO for image processing is basic and uncomplicated, because the system rearranges the input picture to be (448×448 pixels), then apply a single convolutional network function. Finally generates the thresholds detection based on the model's confidence, as shown in Figure 2.

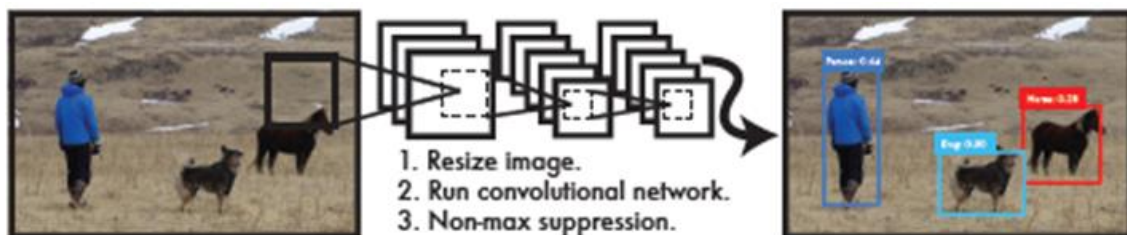


Figure 2. Detection strategy of YOLO

## 2.1. Network design

The PASCAL VOC detection dataset [11] was used to test this design, which was constructed as a convolutional neural network. The visual features are extracted by the network's early convolutional layers, while the fully connected layers predict the output probabilities and coordinates. The network architecture is based on the Google Net image classification model [12]. It contains 24 convolutional layers, followed by two fully connected layers. Instead of Google Net initiation modules,  $1 \times 1$  downsampling layers followed by  $3 \times 3$  convolutional layers were used, as done by Lin *et al.* [13]. Figure 3 depicts the entire network.

A fast version of YOLO trained and developed to push the boundaries of fast object detection fast YOLO uses a neural network with fewer (9 vs 24) convolutional layers and filters in those layers. All training and testing settings are the same between YOLO and fast YOLO, except for the network size. The network's overall output is the  $7 \times 7 \times 30$  tensor of predictions.

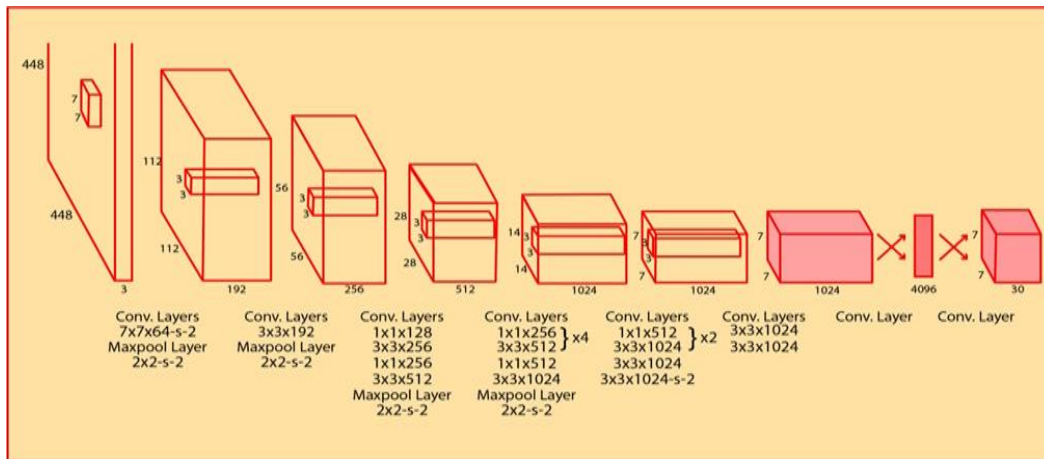


Figure 3. YOLOv2 architecture

## 3. RESEARCH METHOD

The proposed technique uses a deep learning discovery network instead of MRI preprocessing techniques. Since the diagnosis of Alzheimer's disease is based on the study of hippocampal atrophy, this method for detection and classification of the hippocampus, as well as the diagnostic strategy. The Alzheimer's disease neuroimaging initiative (ADNI) database was utilized to compile the data for this study. The ADNI easy search tool is used to search and download data from 1000 raw, unprocessed participants (300 each for AD, MCI and NC). The T1-weighted, magnetization prepared rapid gradient-echo (MPRAGE) series was used to create all of the pictures. Only the data from the sagittal projection is chosen for the best display of hippocampus thickness. The MRI data of each of the 1000 individuals contains 166-180 digital imaging and communications in medicine (DICOM) slices per screen session multiple data preparation stages are performed manually in order to obtain the slices that best represent the hippocampal region of their respective categories [14], [15].

Because the diagnosis of Alzheimer's disease is based on the study of hippocampal atrophy, which accounts for may be less than 5% of the overall MRI image. It's obvious that with no these pre-processing techniques to highlight the presence of the hippocampus, the MRI classification would be inaccurate. Hippocampal detection, which locates and classifies the hippocampal region as AD or NC, should be a better technique than classifying the complete MRI image with (95%) percent redundant background information. Furthermore, it was possible to avoid the MRI pre-processing stage, which would have included many standard pre-processing approaches and thus increased the diagnostic complexity. Each MRI image contains 166-180 DICOM slices from patient as an MRI data. Multiple data preparation stages are carried out manually in order to obtain the slices that best represent the hippocampus area of their respective classes. As a result, during first stage the slice (104) of the MRI data is selected as the best representation of hippocampal atrophy [8], for each patient, which shows the thickness of the hippocampus better. In the second stage the hippocampus area of the selected images is labeled using MATLAB'S image labeler manually as shown in Figure 4. It was suitable to select 300 subjects to be converted from medical DICOM images format to PNG -32-bit lossless format. The data set includes 100 labeled middle hippocampal slices for AD and 100 labeled middle hippocampal slices for NC and 100 labeled middle hippocampal slices for MCI.

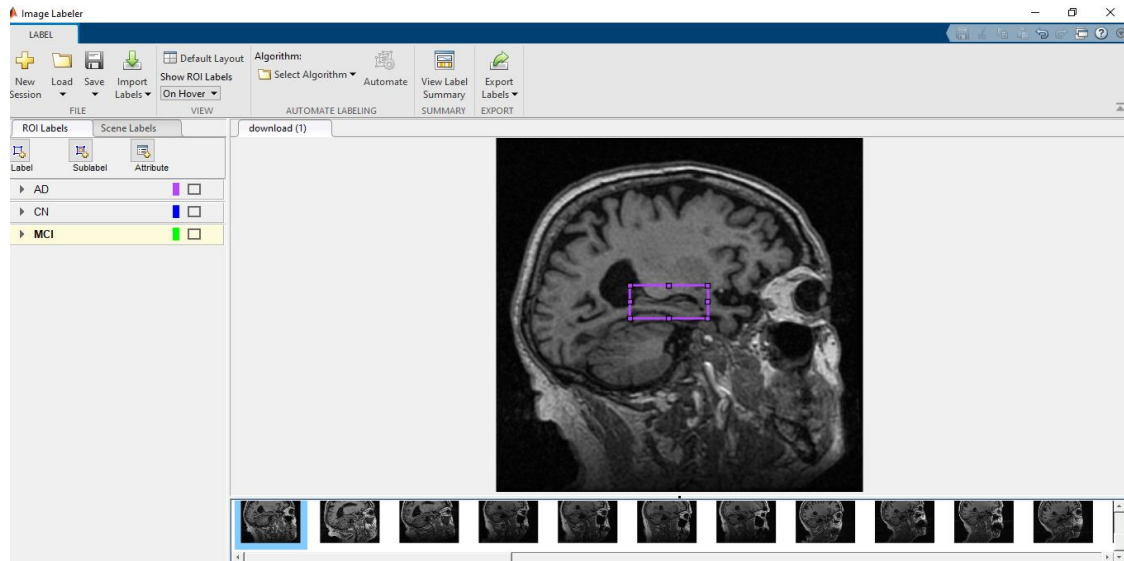


Figure 4. Process of labeling hippocampus

The dataset is saved as `grtruth` files (a ground-truth dataset is an ordinary dataset with annotations added to it.) in the workspace. Annotations can be drawn boxes over photos, written language identifying samples, a new spreadsheet column, or anything else the machine learning program should learn to output). Then in stage three separate the data set into two parts: a training set for the detector and a test set for evaluating the detector. 70% of the data is used for training, while the rest is used to evaluate. The `ImageDatastore` object is used to manage the size of all images, which is  $2048 \times 2048 \times 3$ . If one image file for each image in the set can be placed in a separate memory, but not necessarily the entire collection, the `ImageDatastore` object is used. and can create `ImageDatastore` an object, specify its properties, and then use the `object` function to import and process data using the `imageDatastore` function. Also, the `boxLabelDatastore` is used to create a data store for the bounding box label data, and then use the `read` function to read the bounding box label data from the `boxLabelDatastore` object, this object function returns a cell array with two or three columns. The `combine` object function was used to create the final step before training a datastore that combines the `boxLabelDatastore` object with an `ImageDatastore` object. The combined datastore was then used to train object detectors using training functions such as `trainYOLOv2ObjectDetector`. Set the feature extraction network and feature layer network. Transfer learning can be used in this case if the data samples are not large, especially if there is no model available in this specific domain. It uses previously trained models' knowledge to train newer models, as well as speeding up the training process and improving overall performance. That is why a pre-trained model is used, by replacing the original classifier with a new classifier that best suits the work needs. Figure 5 depicts the concept of transfer learning.

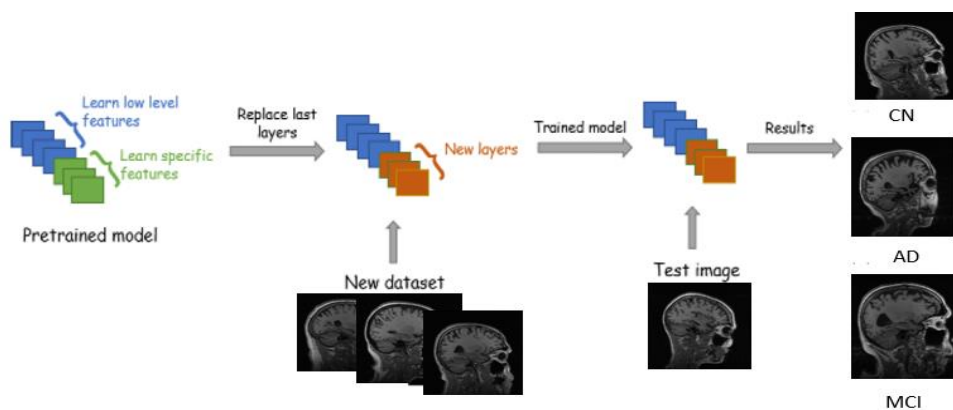


Figure 5. Transfer learning concept

ResNet50 is a pre-trained model that has been trained on more than 1 million images and can categorize them into 1000 different categories, it starts with loading the pre-trained model, which extracts features through convolutional layers (prev. layers). The input image is then classified using the final fully connected layer and the final classification layer. The number of classes in the fully connected layer is replaced by a new one with the same number of outputs as the number of classes in the new data set. The classes of network outputs are defined by the classification layer, which is replaced by a new classification layer without class labels. The output classes of the classification layer will be assigned automatically when the new data set is trained. The YOLOv2 object detection network consists of two sub-networks. A feature extraction network followed by a detection network. The feature extraction network, in this work is a ResNet-50. It has 50 residual layers and an "identity shortcut connection" that skips block of convolutional layers in order to create residual blocks. These remaining blocks address the problem of training accuracy degradation in deep networks, compared to the feature extraction network, the detection sub-network is a small CNN, which consists of several convolutional layers and YOLOv2-specific layers, the convolution layer's main goal is to predict object class probabilities, as well as x and y location offsets, width, and height offsets for each anchor box. The transformation layer extracts the last activations of the convolutional layer and transforms the bounding box predictions to within the bounds of the ground truth. It improves network stability by restricting location predictions. On the other hand, the output layer provides the returned bounding box locations for the target objects [15].

The feature extraction layer, which represents the YOLOv2 network's input, was chosen to be "activation 40 relu," which represents the ReLU layer beyond the 13th building block. The YOLOv2 network was made up of two convolution, batch normalization, and rectified linear unit (ReLU) layers that were repeated. establish yolo, the image size, number of classes, anchor box size, base network name, and feature layer were all passed into the YOLOv2 layer function, which was used to create the network, start training the trainYOLOv2ObjectDetector function was used to train the detector. As a solver, the "sgdm" optimizer was used, with an initial learning rate of 1e4 and a training period of 350 epochs. On a single central processing unit, the total training time for 300 images was 2 hours. The detector was saved to the workspace at the end of the training [16]. This could then be saved as a detector and applied to the test image datasets for validation and testing. Set the maximum number of epochs for training to 20, and use a mini-batch with 16 observations at each iteration [17], [18].

#### 4. RESULTS AND DISCUSSION

YOLOv2 deep learning object detector is used in this study. The dataset is in PNG-32-bit format, with an input dimension of 224×224×3, and is split into a 7:1 ratio between training and test datasets. In order to speed the training process.

A total of 300 images were used, 100 of which were demented and 100 of which were not and 100 in the last stage of Alzheimer's disease. It was decided to choose the training data set to be 75% of the available number of MRI data, and the testing data set to be 25% to evaluate the performance of the algorithm. The accuracy obtained by this procedure was 98 percent. Figures 6, 7, and 8 show the results in graph form.

There are various criteria for evaluating the performance, including recall, precision, and accuracy. Precision is important because it indicates how many of the expected positive values are in fact positive, which is useful if it is wanted to be confident in our forecast as (2) [19].

$$Precision = \frac{True\ positive}{True\ positive + False\ positive} \quad (2)$$

Recall (sensitivity) is another extremely useful parameter that may be used to determine, for example, the proportion of successfully labeled positive values in the overall positive values, as (3) [20].

$$Recall = \frac{True\ positive}{True\ positive + False\ negative} \quad (3)$$

Accuracy is defined as the percentage of correct predictions (both true negative and true positive) out of the total number of cases studied as (4) [15], [21].

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ positive + True\ negative + False\ positive + False\ negative} \quad (4)$$

The value 0.72629 is shown in Figure 9 is a result of finding one case of Alzheimer's disease. The model produces incredibly amazing predicted results, as seen in Figure 10. The model predicts an item based

on its probability [22], [23]; if the probability is less than a specific threshold (here, 0.3), the model predicts that the object does not have Alzheimer's disease [24], [25]. An object's score is a calculation that determines whether or not an object will appear in the expected box. The model's confidence score indicates how certain it is that the box contains an object, as well as how accurate the predictions are in predicting the box. This research achieved a testing accuracy of 0.998 for YOLOv2 without using any MRI pre-processing techniques on the dataset [26], and the proposed method using detection on raw MRI scan is far simpler and clearer while maintaining excellent implication performance.

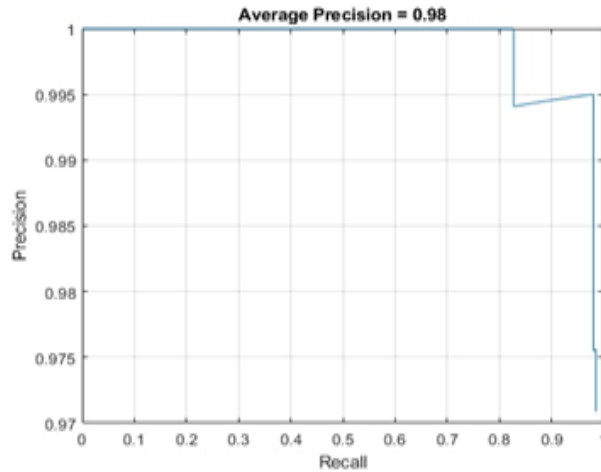


Figure 6. Recall and precision graph

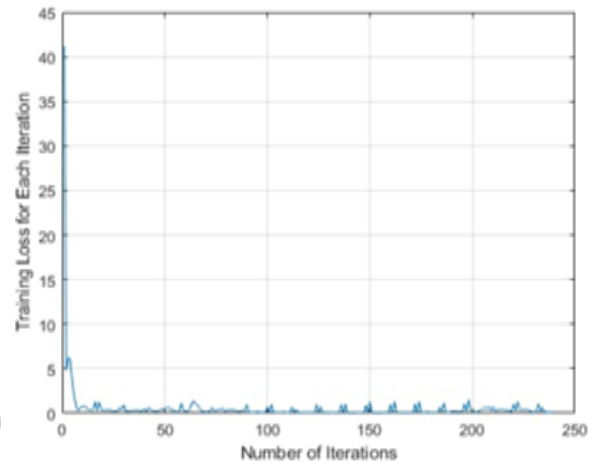


Figure 7. Training loss versus iterations

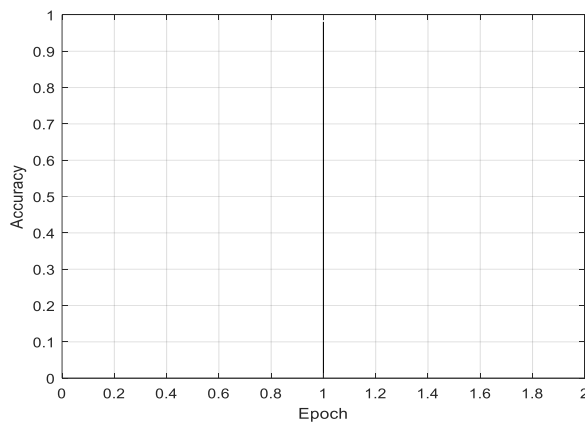


Figure 8. Final accuracy of the method used

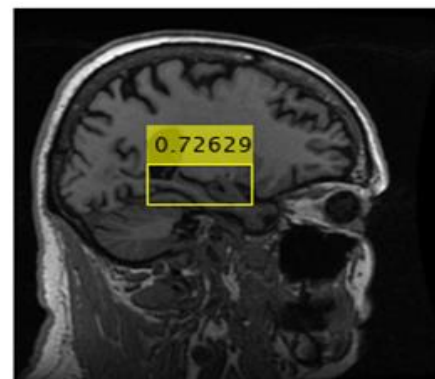


Figure 9. Detection result

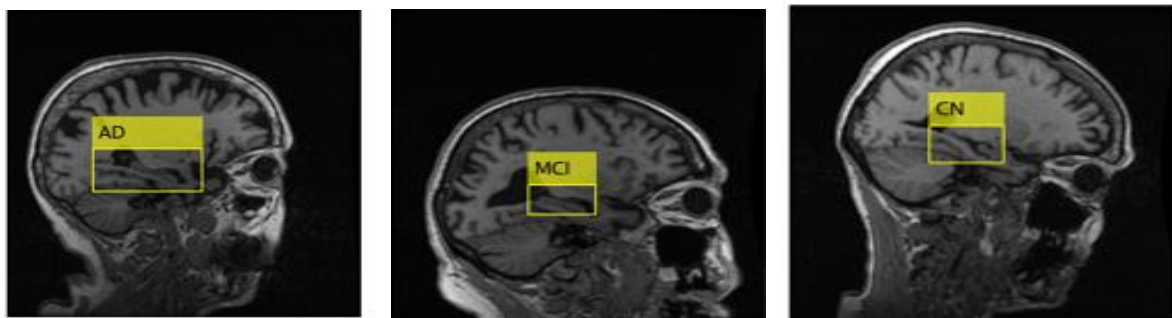


Figure 10. Classification results for the three classes early (AD), mild (MCI), and normal (CN)

## 5. CONCLUSION

In this study, YOLOv2 deep learning object detection architectures were applied to diagnose Alzheimer's disease, and were able to avoid the MRI pre-processing phase, which would have changed the contents of the MRI slides and made the diagnosis more difficult. Also, a new AD/NC object definition standard dataset from deep learning was presented, it includes 200 raw, unprocessed check states for each category, as well as annotated box data, for further object detection study. The YOLOv2 deep network architecture was tested for its ability to detect Alzheimer's disease in a variety of patients. The YOLOv2 method outperforms current strategies in terms of accuracy and real-time performance, especially in complicated settings, according to experimental results on the basic MRI image dataset. YOLOv2 is an excellent choice for Alzheimer's disease MRI detection. Mild cognitive impairment (MCI), the intermediate category between AD and NC, has two stages early and late may be submitted to the data set in the future for classification. It would also be possible to predict Alzheimer's disease by segmenting the hippocampus without using pre-processing of the MRI images.

## REFERENCES




- [1] R. Brookmeyer, E. Johnson, K. Ziegler-Graham, and H. M. Arrighi, "Forecasting the global burden of Alzheimer's disease," *Alzheimer's & Dementia*, vol. 3, no. 3, pp. 186–191, Jul. 2007, doi: 10.1016/j.jalz.2007.04.381.
- [2] A. J. Dinu, R. Ganesan, F. Joseph, and V. Balaji, "A study on deep machine learning algorithms for diagnosis of diseases," *International Journal of Applied Engineering Research*, vol. 12, no. 17, pp. 6338–6346, 2017.
- [3] S. Sarraf, D. D. DeSouza, J. Anderson, G. Tofighi, and A. D. N. Initiativ, "Tofighi deepad: Alzheimer's disease classification via deep convolutional neural networks using MRI and fMRI," *bioRxiv*, 2017, doi: 10.1101/070441.
- [4] W. H. Organization, "World report on ageing and health," 2015. [Online]. Available: <https://apps.who.int/iris/handle/10665/186463>.
- [5] T. H. H. Aldhyani, A. S. Alshebami, and M. Y. Alzahrani, "Soft clustering for enhancing the diagnosis of chronic diseases over machine learning algorithms," *Journal of Healthcare Engineering*, vol. 2020, pp. 1–16, Mar. 2020, doi: 10.1155/2020/4984967.
- [6] D. K. Bon *et al.*, "Clinical practice guideline for dementia by clinical research center for dementia of South Korea," *Journal of the Korean Medical Association*, vol. 54, no. 8, pp. 861–875, 2011, doi: 10.5124/jkma.2011.54.8.861.
- [7] K. H. Thanoon, S. Q. Hasan, and O. I. Alsaif, "Biometric information based on distribution of Arabic letters according to their outlet," *International Journal of Computing and Digital Systems*, vol. 9, no. 5, pp. 981–991, Sep. 2020, doi: 10.12785/ijcds/090518.
- [8] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International Journal of Computer Vision*, vol. 111, no. 1, pp. 98–136, Jan. 2015, doi: 10.1007/s11263-014-0733-5.
- [9] A. J. Dinu, R. Ganesan, F. Joseph, and V. Balaji, "Quality analysis of various deep learning neural network classifiers for Alzheimer's disease detection," *Journal of Engineering and Applied Sciences*, vol. 12, no. 8, pp. 8334–8339, 2017, doi: 10.3923/jeasci.2017.8334.8339.
- [10] S. G. Mueller *et al.*, "The alzheimer's disease neuroimaging initiative," *Neuroimaging Clinics of North America*, vol. 15, no. 4, pp. 869–877, Nov. 2005, doi: 10.1016/j.nic.2005.09.008.
- [11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.
- [12] S. Q. Alhashmi, K. H. Thanoon, and O. I. Alsaif, "A proposed face recognition based on hybrid algorithm for features extraction," in *2020 6th International Engineering Conference "Sustainable Technology and Development" (IEC)*, Feb. 2020, pp. 232–236. doi: 10.1109/IEC49899.2020.9122911.
- [13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 580–587. doi: 10.1109/CVPR.2014.81.
- [14] I. A. Saleh, W. A. Alawsi, O. I. Alsaif, and K. Alsaif, "A prediction of grain yield based on hybrid intelligent algorithm," *Journal of Physics: Conference Series*, vol. 1591, no. 1, Jul. 2020, doi: 10.1088/1742-6596/1591/1/012027.
- [15] M. Lin, Q. Chen, and S. Yan, "Network in network," 2014. doi: 10.48550/arXiv.1312.4400.
- [16] J. X. Fong, M. I. Shapiyai, Y. Y. Tiew, U. Batool, and H. Fauzi, "Bypassing MRI pre-processing in Alzheimer's disease diagnosis using deep learning detection network," in *2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA)*, Feb. 2020, pp. 219–224. doi: 10.1109/CSPA48992.2020.9068680.
- [17] A. M. Khidhir, B. A. Mustafa, and N. T. Saleh, "Design and implementation of a CPU bound process migration in windows 7," in *2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT)*, Nov. 2012, pp. 360–365. doi: 10.1109/ACSAT.2012.30.
- [18] Z. Oleiwi, K. Thanoon, and K. Alsaif, "High frequency coefficient effect on image based on contourlet transformation," in *2019 International Conference on Computing and Information Science and Technology and Their Applications (ICCISTA)*, Mar. 2019, pp. 1–4. doi: 10.1109/ICCISTA.2019.8830649.
- [19] I. A. Saleh, O. I. Alsaif, and M. A. Yahya, "Optimal distributed decision in wireless sensor network using gray wolf optimization," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 9, no. 4, pp. 646–654, Dec. 2020, doi: 10.11591/ijai.v9.i4.pp646-654.
- [20] D. M. Ali, "Health effects of ultra high magnetic fields (MRI as case study)," *International Journal of Computing and Network Technology*, vol. 7, no. 1, pp. 21–27, 2019. doi: 10.12785/ijcnt/070104.
- [21] C. Szegedy *et al.*, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, vol. 40, no. 1, pp. 1–9. doi: 10.1109/CVPR.2015.7298594.
- [22] S. A. Mohamed, O. I. Alsaif, and I. A. Saleh, "Intrusion detection network attacks based on whale optimization algorithm," *Ingénierie des systèmes d'information*, vol. 27, no. 3, pp. 441–446, Jun. 2022, doi: 10.18280/isi.270310.
- [23] A. H. Maray, O. I. Alsaif, and K. H. Thanoon, "Design and implementation of low-cost medical auditory system of distortion otoacoustic using microcontroller," *Journal of Engineering Science and Technology*, vol. 17, no. 2, pp. 1068–1077, 2022.






- [24] M. L. Muammer and O. Alsaif, "Employing X-ray images characteristics for early corona virus detection based on deep learning techniques," *NeuroQuantology*, vol. 20, no. 8, pp. 5664–5673, 2022, doi: 10.14704/nq.2022.20.8.NQ44594.
- [25] H. I. Alsaadi, R. M. Almuttairi, O. Bayat, and O. N. Ucani, "Computational intelligence algorithms to handle dimensionality reduction for enhancing intrusion detection system," *Journal of Information Science and Engineering*, vol. 36, no. 2, pp. 293–308, 2020, doi: 10.6688/JISE.202003\_36(2).0009.
- [26] D. M. Ali and A. S. Mahmoud, "Internet of things security assessment in healthcare environment," *2019 International Conference on Advances in the Emerging Computing Technologies (AECT)*, 2020, pp. 1–4, doi: 10.1109/AECT47998.2020.9194216.

## BIOGRAPHIES OF AUTHORS






**Sara Saad Abd Al-jabar**    received her B.Sc in Computer Engineering from Technical Engineering Collage of Mosul at 2018. Also received M.Sc from the same collage at 2022. Current she is an employee at industrial bank. Her research interests include image processing and microelectronic. She can be contacted at email: saraalbadrany@gmail.com.



**Nasseer Moyasser Basheer**    received his B.Sc. in electrical engineering from University of Mosul, Iraq, at 1980. Also received M.Sc. and Ph.D from the same college at 1984 and 2008. Currently an assistant professor in the Technical Engineering College-Mosul/ Northern Technical University. The research interests include image processing and compression, FPGA applications, and microcontroller applications. He can be contacted at email: nmbasheer@ntu.edu.iq.



**Omar Ibrahim Alsaif**    received his B.Sc. in electrical engineering from University of Mosul, Iraq, in 1992. He received his M.Sc. and Ph.D degrees in Electronics and Microelectronic Engineering from Mosul University, in 2005 and 2018 respectively. He is currently a lecturer in the Mosul Technical Institute/ Northern Technical University in Mosul/ Iraq. His research interests include microelectronic and solid-state systems, renewable energy and nanotechnology devices. He can be contacted at email: Omar.alsaif@ntu.edu.iq.