Online-Journals.org

iJOE International Journal of Online and Biomedical Engineering

iJOE | eISSN: 2626-8493 | Vol. 19 No. 12 (2023) | OPEN ACCESS

https://doi.org/10.3991/ijoe.v19i12.42607

PAPER

# A Comprehensive Study of Deep Learning and Performance Comparison of Deep Neural Network Models (YOLO, RetinaNet)

**Nadia Ibrahim Nife**[1,2](✉), **Mohamed Chtourou**[2]

[1]University of Kirkuk, Kirkuk, Iraq

[2]Control & Energy Management Laboratory, National School of Sfax Engineers (ENIS), University of Sfax, Sfax, Tunisia

nadia.ibra@uokirkuk.edu.iq

**ABSTRACT**

This paper presents the latest advances in machine learning techniques and highlights deep learning (DL) methods in recent studies. This technology has recently received great attention as it can solve complex problems. This paper focuses on covering one of the deep learning algorithms (deep neural network) and learning about its types such as convolutional neural network (CNN), Recurrent Neural Networks (RNN), etc. We have discussed recent changes, such as advanced DL technologies. Next, we continue analyzing and discussing the challenges and possible solutions of machine learning, such as big data and object detection, studying more papers in deep learning, and knowing the main experiments and future directions. In addition, this review also identifies some successful deep learning applications in recent years. Moreover, in this paper, one of the deep learning methods, convolutional neural networks, is applied to detect objects in images by using the You Only Look One model and comparing it with RetinaNet using pre-trained models. The results found that CNN (using YOLOv3) is a more accurate model than RetinaNet.

**KEYWORDS**

deep learning, neural networks, big data, object detection, convolutional neural network (CNN), YOLO model, RetinaNet model

## 1 INTRODUCTION

With the significant advances in technology development of algorithms, artificial intelligence technology occupied an important place in recent years as it began to spread in all aspects of life. DL is the future technology as it considers one of the basic machine learning algorithms, as shown in Figure 1. Machine learning permits systems to develop through learning from the experiments of others [1]. It was developed as a powerful tool for identifying patterns and relationships in data. In addition, it is characterized by flexibility and the ability to adapt to problems [2]. Deep learning models

Nife, N.I., Chtourou, M. (2023). A Comprehensive Study of Deep Learning and Performance Comparison of Deep Neural Network Models (YOLO, RetinaNet). *International Journal of Online and Biomedical Engineering (iJOE)*, 19(12), pp. 62–77. https://doi.org/10.3991/ijoe.v19i12.42607

Article submitted 2023-04-26. Resubmitted 2023-06-25. Final acceptance 2023-06-27. Final version published as submitted by the authors.

require access to large amounts of training data and processing power. Additionally, the combination of standard and deep features also affects achievement efficiency [3].

Deep learning lacks efficiency when trained with an extensive training sample [4]. DL uses new algorithms based on artificial intelligence and simulation of neurons in the human body called the neural network. In recent years, neural networks have received much attention from specialists and researchers. The model trained using a large set of labeled data and neural network structures with multiple layers. This algorithm builds to learn features without defining those characteristics and creates accurate predictive models from unstructured data. Deep learning processes information similar to the human brain and is used in medical fields, face recognition, translation, big data analytics (such as images), natural language processing (NLP), and the stock market. This paper provides an overview of deep learning algorithms and applications and the challenges and problems of applying deep learning in Big Data. Big data are speedily growing in all engineering fields [5]. DNNs have indeed revolutionized the field of computer vision, especially with the advent of novel deeper architectures such as residual and Convolutional Neural Networks [6]. In addition to images, sequential data such as text and audio also are processed using DNNs to reach the documents' most up-to-date classification and recognition performance. Many researchers utilize deep neural networks (DNNs) because of their better performance and fast execution at test time [7]. Deep learning approaches analyze big data with successful applications in speech recognition and computer vision [8]. In machine learning, the features of objects are find manually by relying on humans, while deep learning takes pictures of objects and extracts their features, as shown in Figure 2. This research covers many aspects of object discovery and reviews primary machine learning methods. It also covers the latest types of deep neural networks (DNN), provides a brief description of each model, deep learning features, and an overview of CNN. Modern deep learning models rely on CNN. Deep convolutional neural networks perform remarkably well on many Computer Vision tasks [9]. However, these networks rely heavily on large data sets to help avoid overfitting [10]. Object detection seeks to locate object instances from many predefined categories in natural images; it has many applications in the real world, such as video surveillance, self-driving cars, etc. [11]. Specifically, this paper provides a comprehensive study of the recent achievements in this field brought about by deep learning techniques and the characteristics of Data Augmentation, such as the amount of training data. The last section is the conclusion of this paper. Figure 1 shows the relationship between AI, ML, DL, and NN.

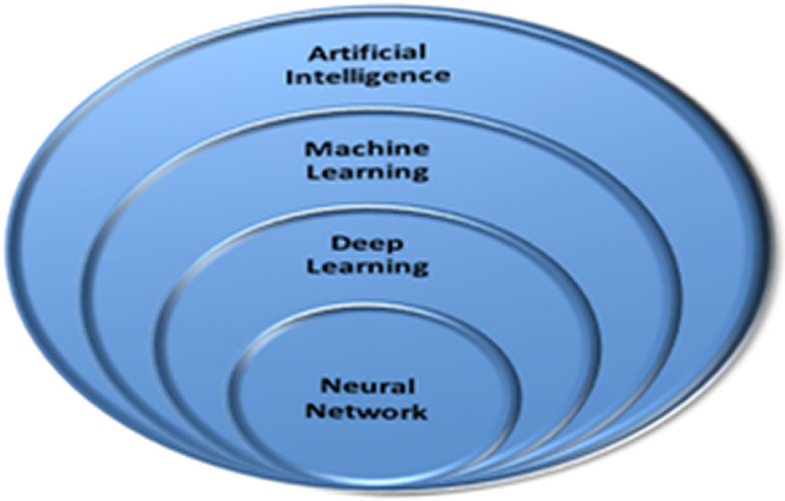


**Fig. 1.** Relationship between AI, ML, DL, and NN


Machine learning process


Deep learning process

**Fig. 2.** Difference between ML and DL

## 2 WORKING MECHANISM FOR DNN WITH DL TECHNOLOGY

Deep learning is a particular branch of machine learning [12]. The latest developments in this field have focused on hands-on training and neural network models, employing algorithms that support deep learning. The diversity of the world's pictures is a significant challenge in recognizing objects [13].

The deep learning mechanism identifies the main problem, collects the data corresponding to the problem, trains this data set using the appropriate deep learning algorithm, and then tests this data. When making an application in deep learning, it builds the features specified without supervision, called unsupervised learning, with speed and accuracy characterizing it, as shown in Figure 3. As in image recognition, its label will describe each image when fetching the data set. Deep learning takes pictures of those objects and finds the properties and differences in them instead of manually finding the properties of those images. Each layer observes a specific pattern in the image. For example, the first layer may notice the image's borders and another layer may mark the characteristics of the objects in the image. A neural network assigns a weight to each neuron, representing the network's information to solve problems. The final layer combines these weighted inputs to come up with an answer. These images feed into the neural network for image recognition, which turns into data that moves between neurons—finally, the last layer groups all these pieces of information to reach a result. Therefore, each picture compares its answer with what the person described. In addition, if the outcome of this network is different from the human response, the neural network modifies the weight of the neurons in the network. Every time it adjusts the weight of the neurons, this improves the ability to recognize images of objects. The neural network stores the practical knowledge to make it available to the user by adjusting the weights. It is one of the most famous algorithms used in machine learning and deep learning, and it is a fast and easy network. It provides multiple training algorithms, as it works well on image, audio, and text data and can be easily updated using new data. Humans supervise a description of each image to identify the object without describing the objects with the neural network. This technology is termed supervised learning [14]. In unsupervised learning, the use of images is without explanation. Here, the neural network must recognize the different patterns in the image to begin recognizing any image containing these patterns. Another example of deep learning is that neural networks can learn words by storing vast amounts of text as they count the number of words they can predict or guess through the terms before or after them. Thus, the program learns to display each word by indicating the relationship of the words with

each other. Sometime, the program reaches and understanding of the terms by itself. Therefore, deep neural networks are constantly changing and developing. They are learnable and change their data analysis, which leads to avoiding previous mistakes. The DL must work with enormous amounts of training data to achieve high accuracy.

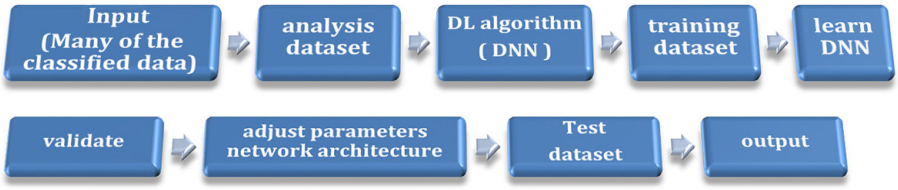


**Fig. 3.** Mechanism of DL and DNNs

## 3 FAMOUS DEEP NEURAL ARCHITECTURE

The achievement of DL and DNN algorithms depends on the design of the network architecture [15]. DNN structures have advanced in many fields, such as image object recognition. We will look more at the significant architectures of DNN. Moreover, we will illustrate the evolution of these deep architectures based on these deep networks. We will focus on CNNs for future understanding and application in image recognition.

### 3.1 RNN. Recurrent Neural Networks

It is one of the structures of deep learning, where the previous inputs are preserved and fed into the feedback of this network, which can be a hidden layer, an output layer, or a combination of both. It depends on feeding the inputs or hidden layers to the output, such as in prediction applications, translation, and converting audio to text. This network is widely used in NLP translation and time series languages [16]. The problem of vanishing gradient occurs in this network. The gradients often disappear in the process of training NNs. The gradients are either too large or too small. To eliminate this problem, we use LSTM for these repetitive networks [17].

**Long Short-Term Memory (LSTM).** LSTM is an RNN used for various applications such as phones. An LSTM consists of three gates controlling the information flow inside or outside the neuron. The input gate contains information entered into the memory, and the forget gate allows new data to remember. The output gate outputs the data from the cell. LSTMs are used in the fields of speech and handwriting recognition and can handle complex detection tasks [18].

**Gated Recurrent Units (GRU).** It is a type of RNN like LSTM in its operation but is more straightforward and characterized by faster implementation and training. It uses smaller redundant data, so it shows better performance. The GRU consists of two gates. Update the portal that offers how much of the previous cell contents should keep. Reset Portal Sets the new entry to merge with the contents of the previous cell. The GRU lacks an output gate.

### 3.2 CNN. Convolutional Neural Networks

CNN is one of the types of deep learning DNN. This NN uses image recognition applications. It made significant progress in this area as it can recognize faces, people,

street signs, and many other places, and apply it in natural language processing and audio and video analysis. CNN can take a picture, enter the image into the network, and distinguish people in it. This network is multilayered because it consists of input and output layers and multiple hidden layers. Hidden layers consist of convolutional layers and use a mathematical model. The results were transferred to successive layers. These layers are the basis for building a convolutional neural network. CNN contains a Relu layer, a Pooling layer, and is fully connected. These technologies use algorithms for artificial intelligence and deep learning from data.

The principal work of this algorithm is as follows.

- The dataset enters the network, and then the image enters and extracts his features.
- Then, the image travels through the convolutional layer, clustering layer, and fully connected layer, in which all neurons are connected to all activation data in the previous layer.
- Finally, we get the output. As the output layer describes the image content, the convolutional and pooling layers are the basis for building this algorithm.

Figure 4 describes the CNN structure. Many concepts enhance CNN, such as activation and loss function, architecture innovation, modulation, and parameter optimization. There are many differences in the CNN architecture based on the layering pattern. Convolutional layers convert the input data from the input layer by the neurons connected with the previous layer into a set of patterns provided by the output layer to activate the neurons with some identifiable tasks. This network has shown an outstanding performance in object detection, as it can extract features by convolution and automatically recognize representations of data [19]. Therefore, the first layers identify the characteristics, while the subsequent layers capture the finer details and organize them in complex properties and shapes. Moreover, the last layer of the NN connects to the neurons in the previous layer. In addition, you put all these characteristics together to classify the image accurately and recognize the object. CNN relies a lot on big data to avoid overuse because CNN will be more accurate as training data increases.

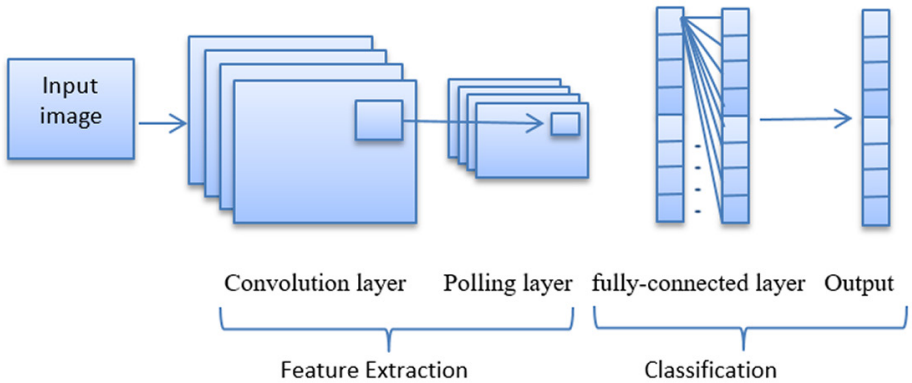


**Fig. 4.** CNN algorithm structure

**The architecture of Deep CNNs.** A convolutional neural network aimed to detect visual objects. The ImageNet project runs a yearly competition called the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). The Convolutional Neural Network (CNN) got perfect results in computer vision tasks [20].

- LeNet. It consists of several layers that extract features from the input image. Then the image is classified and the extracted features are grouped and convolution

performed. The output layer in this grid is a group of cells that define the characteristics of an image.
- AlexNet. It is one of the deepest CNN structures used in computer vision.
- ZF Net. These deep browns are used in deconvolution.
- GoogleNet. They are deep and robust nervous structures that were designed by Google.
- SqueezeNet. This architecture is robust and functional in mobile systems.
- SegNet. This structure is used for image segmentation.
- VGGNet. These structures are hierarchical, and the layers at the bottom are broad, while the layers at the top are deep.
- YOLO (You only look once). It discovers images by dividing them into squares and determining the object's class for each section. This network is the most famous and used to develop deep structures today. The object detection model (YOLO) has resulted in a range of versions based on continuous improvements to this model in computer vision. (Yolo, Yolov2, Yolov3, Yolov4, Yolov5, Yolov6, Yolov7, Yolov8, and currently Yolo-). Thus, the most urgent requirement for improving object detection is acceleration. One of the best CNN models is YOLO, which breaks the speed limit of CNN. YOLO achieves a significant balance between speed and accuracy and is an object detector with a strong generalizing ability to perform the whole image.
- ResNet. These are deep learning constructs using shortcuts.

Table 1 [20], shows a general summary of the different models for the architecture of a deep convolutional neural network.

**Table 1.** ILSVRC competition CNN models

| Year | CNN | Developed by | Place | Top-5 Error Rate | No. of Parameters |
|---|---|---|---|---|---|
| 1998 | LeNet | Yann LeCun et al. | | | 60 thousand |
| 2012 | AlexNet | Alex Krizhevsky, Geoffrey Hinton, llya Sutskever | 1st | 15.3% | 60 million |
| 2013 | ZFNET | Matthew Zeiler and Rob FerGus | 1st | 14.8% | |
| 2014 | GoogleNet | Google | 1st | 6.67% | 4 million |
| 2014 | VGG Net | Simonyan, Zisserman | 2nd | 7.3% | 138 million |
| 2015 | ResNet | Kaiming He | 1st | 3.6% | |

## 4 TRAINING OF DEEP NEURAL NETWORKS

Training occurs in the deep neural network by recognizing different data and comparing it with the expected results. Through a reverse process of directing the data from the output to the input layers, comparing and correcting errors to reach the desired result.

Deep neural network training occurs in several stages, which is as follows.

1. In this stage, using the DNN architecture, the number of I/O and hidden layers are determined. Determining the number of input units extracts the number of data features used in training. During the training process, the algorithms discover valuable data [21]. As for the hidden layers, the number of units used in them should be the same in each hidden layer. In the output layer, the number of output units that

the neural network wants to process is determined. The input layer is associated with the hidden layers attached to the output layer through neural connections.

As shown in the example in Figure 5 below, a deep network contains input and output layers and many hidden layers. Assume that the input layer of the network is x, as the neurons input x1 to xn, and the output layer is y between each layer.

2. Initialization. There is a set of actual parameters (weights) from W1 to Wn. Likewise, each layer consists of biases b that change the product's weight multiplied by the input. This stage assigns the initial weights W and bias b to the inputs of all neurons and random numbers between (0) and (1). Weights are updated during the training process to suit the network tasks.

   For example, as shown in Figure 6 we adjust the weights gradually to train deep neural networks efficiently until we get excellent predictions. Pass the input through the neural network by the forward propagation algorithm. Then the weights w1, 2, 3, 4 are multiplied in the first layer, the bias is added, the activation function is applied, and the result is multiplied by the weights of the second layer, then the result is calculated. In addition, multiply the weights by the corresponding inputs and then add to the bias, and this value is called K, K = ΣW * X + b. The activation function f is chosen for each hidden layer, where the activation function applied to K and f (K) is the final output of the neuron.

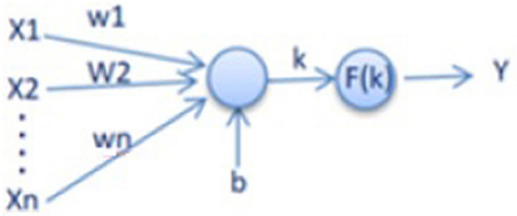


**Fig. 5.** A deep neural network model

3. Select the optimization algorithm and determine the suitability of the DNN for the training data.
4. Error function. It is the correct output value as we subtract the actual network output from the model. The cost function is the square of the average error and is given as equation (1).

$$C = 1/m \sum(y - a) \wedge 2 \quad (1)$$

   Where m. is the number of training inputs in the network, a., the expected value, and y., the actual value

5. This stage trains the data input to the network using the backpropagation algorithm that feeds the data into the network and evaluates the network performance.
6. Change the neurons' weights to reduce the error function to a minimum and increase the accuracy.
7. Applying the desired inputs x1, x2, x3, and the selected output y (k) and find the outputs at k = 1, where k is the number of iterations. This is called an activation function and represented as in Equation 2.

$$y = sign \sum_{i=1}^{n} xiwi - \theta \quad (2)$$

*n*. network input

Several activation functions define the network output, Including sigmoid, sign, step, and linear. The best-used sigmoid activation function is in the following equation (3).

$$x = \sum_{i=1}^{n} xiwi \tag{3}$$

$$y = \frac{1}{1 + e^{-x}} \tag{4}$$

Where weighted inputs in the network = $x$

Input value $xi = i$, Weight $wi = i$, Number of neuron inputs = $n$, Neuron output = y

8. Iterations continue until the network learns. Increase iteration $k$ by 1, return to the second step, and repeat the process until the convergence point. The gradient descent force updates the weights with a small loss function after iteration. Calculates the expected output $\hat{y}$, known as feed-forward, at each iteration of the training process. Weights and biases are also updated, known as backpropagation, as shown in equation 5. An example is training a DNN consisting of four layers by adjusting the weights and biases of the data input to the network.

$$\hat{y} = \sigma(w2\sigma(w1x + b1) + b2) \tag{5}$$

We can see from the above equation, the weights $W$ and the biases $b$ are the only variables that affect the output y, as shown in Figure 6.


**Fig. 6.** Training process of deep neural network

At the end of the training process, the model is ready to make predictions of the input data. It can feed the new data, pass it forward, and generate the predicted model.

## 5 PROBLEMS FOR DEEP NEURAL NETWORKS

Despite the significant advances made by deep learning systems on a large scale, there are many challenges to overcome when designing and training deep networks.

- Object detection and localization of all objects in the image, one of the most basic and complex problems in computer vision, determines the states of objects from many predefined classes in natural images. This problem arises in the case of poor data quality and poor resolution and clarity of the images when identifying things, so data quality is essential to obtain accurate results.
- Object recognition is one of the most fundamental and challenging problems of computer vision [22].

– This technology also faces many challenges, including the clarity of images. With the advancement of deep learning research and its applications, we aspire to more exciting research on DL and work to improve data augmentation to take advantage of the potential of big data and increase the time and the size of the final data set [23]. Therefore, network training should be large, and accurate. Also, choosing a good DL architecture is essential to get better results from NNs [24]. There are now many different algorithms for detecting objects. It is challenging to choose the best models for detecting objects. Therefore, in this research, we made a comparison of these models. The importance of object detection is the ability of the implemented structure to determine the type and location of a specific object in an image and the speed of detection. The deep learning algorithm (CNN) is one of the most active areas of research that has achieved success in object detection. In this paper, we used different architectures to detect objects in images and compare them. These architectures include Yolo network architecture and RetinaNet architecture. Used Image AI library in Python, as this library is a database designed for use in object recognition. This library supports several pre-trained detection models for modern deep learning algorithms, such as RetinaNet and YOLOv3, trained on the COCO data-set. For 90 common objects. Object detection determines the name and location of each object in the image and the percentage probabilities of all detected objects.

## 6 EXPERIMENTAL RESULTS

To perform the object recognition using a RetinaNet model, note that Figure 7 represents the original image before detecting the objects. We used one of the remote-sensing images. One of its most important applications is the population census in an area, as it is difficult to identify small objects in this type of image. Therefore, we find in this study the best model for detecting objects accurately and with a high training speed.

**Fig. 7.** Image before object detection

Figure 8 shows output after detecting objects using the RetinaNet model. In Figure 8a represents the resulting image after detecting the objects in it, and

Figure 8b specifies the name of the object and the degree of accuracy of detection using the RetinaNet model. It has very high accuracy and speed.

a)


b)

ODetectionRetinaNet.py

ODetectionRetinaNet.py > detections

1

OUTPUT TERMINAL DEBUG CONSOLE PROBLEMS

```
2023-05-11 22:11:20.477317: W tensorflow/stream_
2023-05-11 22:11:20.477563: W tensorflow/stream_
2023-05-11 22:11:20.479951: I tensorflow/stream_
2023-05-11 22:11:20.480239: I tensorflow/stream_
2023-05-11 22:11:20.480716: I tensorflow/core/pl
To enable them in other operations, rebuild Tens
WARNING:tensorflow:No training configuration fou
2023-05-11 22:11:27.523331: I tensorflow/compile
car  :  70.22340297698975
car  :  67.97776818275452
person  :  64.20820951461792
car  :  60.91098189353943
person  :  60.87021827697754
person  :  60.77108979225159
umbrella  :  60.13510227203369
person  :  56.81872367858887
umbrella  :  56.43693208694458
person  :  55.025821924209595
person  :  53.75070571899414
person  :  52.075183391571045
person  :  51.10286474227905
person  :  50.11441707611084

[Done] exited with code=0 in 15.964 seconds
```

**Fig. 8.** Output after detecting objects using the RetinaNet model: (a) the resulting image after detecting the objects (b) the name of the object and the degree of accuracy of detection

Table 2, shows results of each class of the retina models.

**Table 2.** Detection results of each class using retina

| Class | Precision |
|---|---|
| car | 70.2 |
| car | 68.0 |
| person | 64.2 |
| car | 60.9 |
| person | 60.9 |
| person | 60.8 |
| umbrella | 60.1 |
| person | 56.8 |
| umbrella | 56.4 |
| person | 55.0 |
| person | 53.8 |
| person | 52.1 |
| Person | 51.1 |
| Person | 50.1 |

When implementing the convolutional neural network model (YOLO) as shown in Figure 9, we see that this model successfully identifies the objects in the image more accurately. Figure 9a shows the objects detected in the image, while the Figure 9b shows the names of the objects and the degree of accuracy of their detection.

a)


**Fig. 9.** *(Continued)*

b)

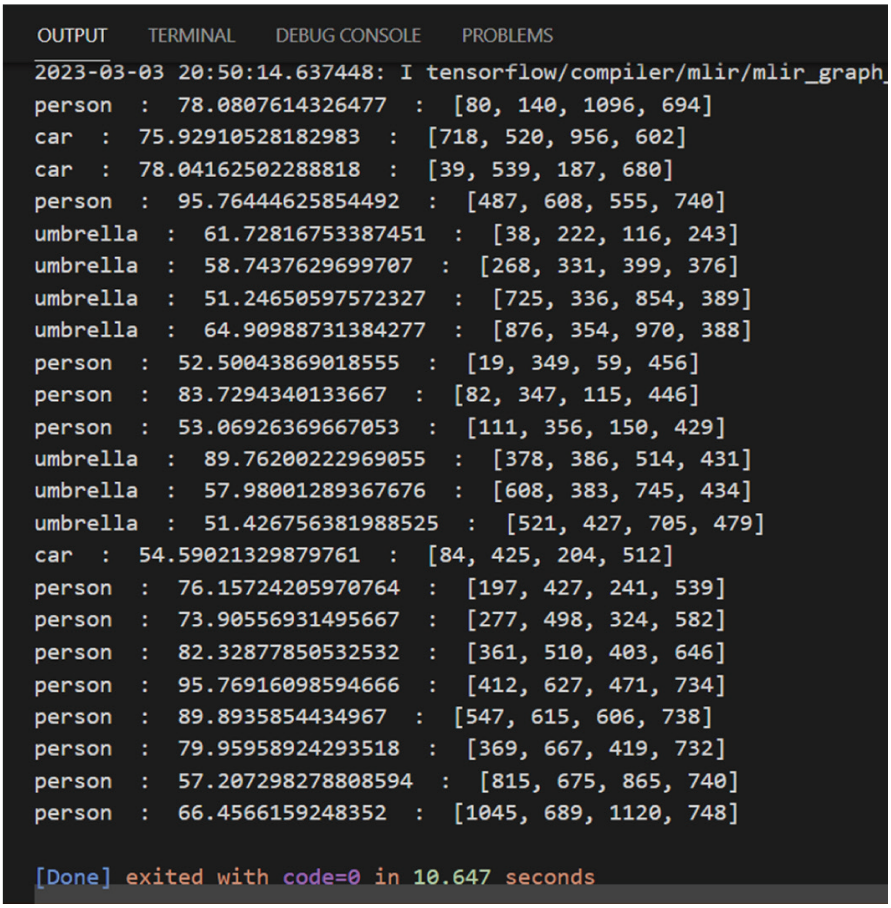


**Fig. 9.** Output of image object detection using the Yolov3 model: (a) Detecting objects in an image (b) Accuracy of detecting objects. Table 3 shows the results of each class of the Yolo models.

**Table 3.** Detection results of each class using Yolo

| Class | Precision |
|---|---|
| person | 78.1 |
| car | 75.9 |
| car | 78.0 |
| person | 95.7 |
| umbrella | 61.7 |
| umbrella | 58.7 |
| umbrella | 51.2 |
| umbrella | 64.9 |
| person | 52.5 |
| person | 83.7 |
| person | 53.1 |
| umbrella | 89.7 |

*(Continued)*

**Table 3.** Detection results of each class using Yolo *(Continued)*

| Class | Precision |
|---|---|
| umbrella | 57.9 |
| umbrella | 51.4 |
| car | 54.5 |
| person | 76.1 |
| person | 73.9 |
| person | 82.3 |
| person | 95.7 |
| person | 89.8 |
| person | 79.9 |
| person | 57.2 |
| person | 66.4 |

Figure 10 displays the number of objects in all classes = 23 in Yolo but 14 in the retina

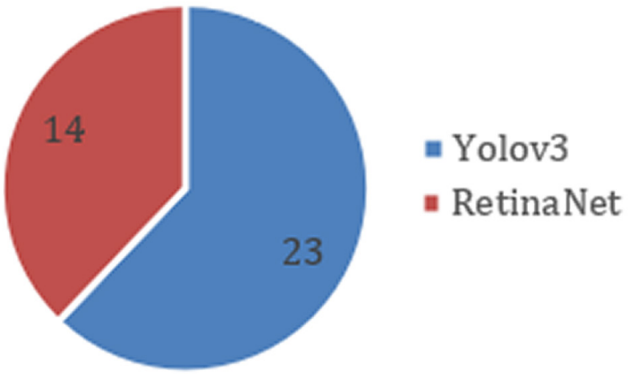


**Fig. 10.** Number of objects in all classes

We observed from the above results in Figures 8 and 9 that Yolo's accuracy and speed are very high compared to RetinaNet's object detection model. Whereas the time taken to detect image objects using the Yolo model is (10.647) seconds. The detection time of objects in RetinaNet is (20.028) seconds.

The ImageAI library has the advantage of being able to extract every object detected in the image. In addition, save each in the newly created folder and return an additional array containing the path of each image.

Yolo engineering also succeeded in classifying the input image into several parts according to the detected objects at a very high speed and specifying the coordinates of each object in the image. Additionally, it defines "box points", meaning, the location of the object's center point, top left point, and bottom right point. It took time to discover and classify the image objects, with the coordinates of those objects determined by (11.432). The results of the accuracy and speed of object detection in the image appear, as shown below in Figure 11.


**Fig. 11.** Outputs of each detected object in the image with its 'box points' using Yolov3

## 7 CONCLUSION

The application of deep learning in various fields has recently become a popular research topic. This study found that the more cases monitored during the model training process, the greater the likelihood of improving prediction results and accuracy. In addition, the increase in the number of hidden layers increases the quality and efficiency of the network, and the time required for it increases. The number of cells for all hidden layers must be equal and be more than the number of features. In addition, in this research, we presented a comparative study of two main neural network models for object detection (RetinaNet and YOLOv3, respectively). YOLO was more accurate, and the average detection times for objects in the image were (20.028) sec. for RetinaNet and (10.647) sec. for YOLOv3, respectively. Therefore, in this paper,

we recommend using YOLO models in object detection applications to obtain the best results. Image detection is an excellent model problem for neural network recognition. It provides a great way to develop advanced deep-learning techniques. In the future, we plan to build a real-time image detection system. Therefore, in this paper's future research, we are improving and developing convolutional neural networks for tracking the visual object in a video. Despite the rapid development and promising progress in detecting objects, there are still many problems for future work.

## 8 ACKNOWLEDGMENTS

Gratitude and appreciation to everyone who helped me complete this research, especially to the Department of Computer Engineering at National School of Sfax Engineers (ENIS), University of Sfax in Tunisia.

## 9 REFERENCES

[1] Nihad, M., Ramadan, F., & Mohammed Ali, S. I. (2023, March). Machine learning methods and approaches for predicting Covid19. In AIP Conference Proceedings (Vol. 2591, No. 1). AIP Publishing. https://doi.org/10.1063/5.0119819

[2] Díez-Sanmartín, C., & Sarasa Cabezuelo, A. (2020). Application of Artificial Intelligence techniques to predict survival in kidney transplantation. A review. Journal of Clinical Medicine, 9(2), 572. https://doi.org/10.3390/jcm9020572

[3] Nadeem, M. S., Franqueira, V. N., Zhai, X., & Kurugollu, F. (2019). A survey of deep learning solutions for multimedia visual content analysis. IEEE Access, 7, 84003–84019. https://doi.org/10.1109/ACCESS.2019.2924733

[4] Xie, L., Zhai, J., Wu, B., Wang, Y., Zhang, X., Sun, P., & Yan, S. (2020, November). Elan. Towards generic and efficient elastic training for deep learning. In 2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS) (pp. 78–88). IEEE. https://doi.org/10.1109/ICDCS47774.2020.00018

[5] Su, Y., Yu, Y., & Zhang, N. (2020). Carbon emissions and environmental management based on Big Data and Streaming Data. A bibliometric analysis. Science of The Total Environment, 733, 138984. https://doi.org/10.1016/j.scitotenv.2020.138984

[6] Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. Artificial Intelligence Review, 53(8), 5455–5516. https://doi.org/10.1007/s10462-020-09825-6

[7] Lavi, B., Serj, M. F., & Ullah, I. (2018). Survey on deep learning techniques for person re-identification task. arXiv preprint arXiv.1807.05284.

[8] Saravana Ram, R., Vinoth Kumar, M., Al-shami, T. M., Masud, M., Aljuaid, H., & Abouhawwash, M. (2023). Deep fake detection using computer vision-based deep neural network with pairwise learning. Intelligent Automation & Soft Computing, 35(2), 2449–2462. https://doi.org/10.32604/iasc.2023.030486

[9] Zhang, L., Wang, S., & Liu, B. (2018). Deep learning for sentiment analysis. A survey. Wiley Interdisciplinary Reviews. Data Mining and Knowledge Discovery, 8(4), e1253. https://doi.org/10.1002/widm.1253

[10] Reddy, K. R., & Dhuli, R. (2023). A novel lightweight CNN architecture for the diagnosis of brain tumors using MR images. Diagnostics, 13(2), 312. https://doi.org/10.3390/diagnostics13020312

[11] Hammoudeh, M. A. A., Alsaykhan, M., Alsalameh, R., & Althwaibi, N. (2022). Computer vision: A review of detecting objects in videos – challenges and techniques. International Journal of Online & Biomedical Engineering, 18(1). https://doi.org/10.3991/ijoe.v18i01.27577

[12] Safie, S. I., & Khalid, P. Z. M. (2023). Practical consideration in using pre-trained Convolutional Neural Network (CNN) for finger vein biometric. iJOE, 19(02), 163. https://doi.org/10.3991/ijoe.v19i02.35273

[13] Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., & Pietikäinen, M. (2020). Deep learning for generic object detection. A survey. International Journal of Computer Vision, 128(2), 261–318. https://doi.org/10.1007/s11263-019-01247-4

[14] Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S. ... & Asari, V. K. (2019). A state-of-the-art survey on deep learning theory and architectures. Electronics, 8(3), 292. https://doi.org/10.3390/electronics8030292

[15] Patel, S., & Patel, A. (2018). Deep learning architectures and its applications. A survey. International Journal of Computer Sciences and Engineering, 6(6), 1177–1183. https://doi.org/10.26438/ijcse/v6i6.11771183

[16] Fawaz, H. I., Forestier, G., Weber, J., Idoumghar, L., & Muller, P. A. (2019). Deep learning for time series classification: A review. Data Mining and Knowledge Discovery, 33(4), 917–963. https://doi.org/10.1007/s10618-019-00619-1

[17] Chan, Y. W., Kang, T. C., Yang, C. T., Chang, C. H., Huang, S. M., & Tsai, Y. T. (2022). Tool wear prediction using convolutional bidirectional LSTM networks. The Journal of Supercomputing, 78(1), 810–832. https://doi.org/10.1007/s11227-021-03903-4

[18] Ait-Bennacer, F. E., Aaroud, A., Akodadi, K., & Cherradi, B. (2022). Applying deep learning and computer vision techniques for an e-sport and smart coaching system using a multiview dataset: Case of Shotokan Karate. International Journal of Online & Biomedical Engineering, 18(12). https://doi.org/10.3991/ijoe.v18i12.30893

[19] Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., & Iyengar, S. S. (2018). A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys (CSUR), 51(5), 1–36. https://doi.org/10.1145/3234150

[20] Siddharth Das. CNN Architectures. LeNet, AlexNet, VGG, GoogLeNet, ResNet and more. Medium, 16 Nov. 2017, https://medium.com/analytics-vidhya/cnns-architectureslenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5

[21] Srivastava, R. K., Greff, K., & Schmidhuber, J. (2015). Training very deep networks. Advances in neural information processing systems, 28, 2377–2385.

[22] Liu, H., Sun, F., Gu, J., & Deng, L. (2022). Sf-yolov5. A lightweight small object detection algorithm based on improved feature fusion mode. Sensors, 22(15), 5817. https://doi.org/10.3390/s22155817

[23] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of Big Data, 6(1), 60. https://doi.org/10.1186/s40537-019-0197-0

[24] Maqsood, S., Damaševičius, R., & Maskeliūnas, R. (2022). Multi-modal brain tumor detection using deep neural network and multiclass SVM. Medicina, 58(8), 1090. https://doi.org/10.3390/medicina58081090

## 10 AUTHORS

**Nadia Ibrahim Nife** received a Bachelor's degree from the Technical College of Kirkuk, Software Engineering Department in 2004. After that, she received a master's degree in the Department of Computer Engineering, at Cankaya University, Ankara, Turkey in 2015. Now a Ph.D. student at the Faculty of Engineering, University of Sfax (Email: nadia.ibra@uokirkuk.edu.iq).

**Mohamed Chtourou** is a professor in the Department of Electrical Engineering of the National School of Engineers of Sfax, Tunisia (Email: Mohamed.chtourou@enis.rnu.tn).