1. Maha Mokrani[1], 2. Zied Hajaiej[2]

University of Tunis El Manar,
Analysis and Processing of Electrical and Energy Signals and Systems Research Laboratory,
Faculty of Sciences of Tunis(1),(2)
ORCID: 1.0009-0007-2433-8996; 2.0009-0002-3255-5249

doi:10.15199/48.2024.05.48

Real time object detection with data variation

***Abstract**. Object recognition has a number of advantages, not least its speed and accuracy. It can be used to identify objects quickly and accurately in real time, and can be used to automate tasks such as security systems. The aim of this paper is a comparative study of the recognition of a specific "knife" object using the yolov8 algorithm. Then we'll see how to train this model ourselves using this dataset. Finally, we'll train Yolov8 to identify custom objects from our own data (photos taken with cameras) and ultimately compare its accuracy.*

***Streszczenie.** Rozpoznawanie obiektów ma wiele zalet, między innymi szybkość i dokładność. Można go używać do szybkiej i dokładnej identyfikacji obiektów w czasie rzeczywistym, a także do automatyzacji zadań, takich jak systemy bezpieczeństwa. Celem niniejszej pracy jest badanie porównawcze rozpoznawania konkretnego obiektu typu „nóż" przy wykorzystaniu algorytmu yolov8. Następnie zobaczymy, jak samodzielnie wytrenować ten model, korzystając z tego zbioru danych. Na koniec przeszkolimy Yolov8 w zakresie identyfikowania niestandardowych obiektów na podstawie naszych własnych danych (**zdjęć wykonanych aparatami) i ostatecznie porównujemy ich dokładność. (Wykrywanie obiektów w czasie rzeczywistym ze zmianami danych**)*

Keywords: Yolov8, Deep learning, Data variation, computer vision.
Słowa kluczowe: Yolov8, uczenie głębokie, zmienność danych, wizja komputerowa.

Introduction

Object detection is a very active field of research which aims to classify and locate regions/areas of an image or video stream.

Indeed, the principle of object detection is as follows: for a given image, we search for regions of it that might contain an object, then for each of these discovered regions; we extract and classify it using an image classification model - for example - . The regions of the original image with good classification results are retained, while the others are discarded. So, to have a good object detection method, it's necessary to have a solid region detection algo and a good image classification algorithm.

The aim of this paper is to compare the results of pre-trained models for image classification and models trained using yolov 8 on images, moving video and in real time.

The paper begins by exploring the fundamental concepts and architecture of the original YOLO model, which paved the way for subsequent advances in the YOLO family. We then examine the refinements and improvements introduced in each version, from YOLOv2 to YOLOv8. These improvements encompass various aspects such as network design, loss function modifications, anchor box adaptations and input resolution scaling. By examining these developments, we aim to offer a holistic understanding of the evolution of the YOLO framework and its implications for object detection. While examining the specific advances of each YOLO version, the article highlights the trade-offs between speed and precision that emerged throughout the mount's development.
Finally, this will present a comparative survey of YOLOV8 compilations of pretraned models and trained models with data variation. [1]

Related work

Object detection is a type of image categorization in which a neural network anticipates elements in an image and draws bounding boxes around them. The detection and localization of elements in an image conforming to a predefined set of classes is called object detection.

Object detection (also known as object recognition) is a particularly important sub-field of computer vision, as tasks such as detection, identification and localization find wide application in real-world contexts.

The YOLO approach can help you accomplish these tasks. In this essay, we'll take a closer look at YOLO, including what it is, how it works, its different variants, etc.

Object recognition can be used in a variety of fields, including automotive, medical and video surveillance and all these fields of application are presented in real time.

YOLOV: General comprehension

Before focusing on YOLO, let's quickly define what detection is in Computer Vision: Given C classes of objects (examples of classes: chair, dog, person, bicycle helmet...), for a given image, if it contains instances of these classes, the aim is to locate and identify the class of these instances. In the YOLO framework, the location of a class instance is done by a bounding box, a rectangle whose sides are horizontal and vertical and which is supposed to frame the detected object as well as possible. The framework is different from classification, which consists in classifying an image in its entirety (is it a cat image or a bicycle image). For an image, we expect an output: a class name/identifier. In the case of detection, the output can be:- Empty (no instance of these C classes detected on the image)-One or more objects for which a bounding box and its class are indicated Example of detection on an image:3 distinct class instances detected.[2]

In the context of image processing, the evolution of technologies has radically changed the approaches and possibilities for understanding the information contained in photos or videos, and doing so automatically.

Locating one or more objects in an image is particularly well handled with the latest technologies available, and framing an object within a rectangular area of an image is now widely achievable.[2] [3]

You Only Look Once (YOLO) is an object detection algorithm renowned for its high accuracy and speed. However, it is the result of many years of research.

During inference, an image is processed only once by a single convolutional network (hereinafter referred to as CNN). To better understand this specificity, let's compare this approach with another detection model in vogue when the first version was conceived

A model like R-CNN decomposes the problem into 2 stages:

- Generate the objects to be detected (potentially using a method).
- Classify areas generated in the previous step and refine them to produce more precise location bounding boxes.

A. The output format and grid division

The neural network's output format, which is a 3D-dimensional tensor, is a common feature among the different versions of YOLO Sx * Sx p.

This corresponds to a slicing of the initial image into a grid of Sx cells. For each of these image portions, the detection results are stored in the p coordinates available per cell.[4]

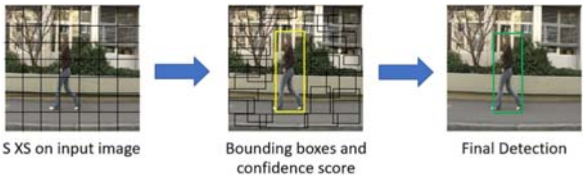

Fig. 1: YOLOV output

From these annotations we need to construct the yi (of dimension S x S x p) for each image which will be used during training. For each object in the image:

- Calculate the coordinates of the bbox center

- Determine the image cell (x, y) where this center is located ($0 \le x, y < S$)

- The coordinates of the bounding box and the object class are thus entered in the column yi[x, y,:]

The other parts of the yi tensor, those corresponding to cells and bounding boxes with no object, are zero.

YOLO treats detection as a regression problem (prediction of bounding box coordinates. Note also that since p is a fixed dimension, the number of objects per cell is limited to a number K. A fortiori, the number of objects detectable by the model on a an image is limited to S x S x K.

YOLO models predict an objectness score for each bounding box, which models the probability that a class instance will be found within the bounding box, times the Intersection over Union (IoU) of this box with the ground truth.

The IoU quantifies the accuracy of the predicted bounding box in relation to the ground truth (annotated data).

$$\text{objectness} = \Pr(\text{object}) . \times \ \text{IOU}\ (bb,gt) \qquad (1)$$

B. Inference

For each predicted bounding box, a confidence score is obtained by multiplying the objectness to the maximum of the class probabilities.

$$\begin{aligned} &\text{IOU}\ (bb,gt) \times \max\ (\Pr\ (\text{Class}_i \mid \text{object})\ ) \\ &= \max\ (\Pr(\text{Class}_i)) \times \text{IOU}\ (bb,gt) \end{aligned} \qquad (2)$$

This score reflects the probability that there is an instance of the class argmax(Pr(Classi)) in the box, as well as its suitability for the object.[7]

If this score is above a certain threshold set for inference (often defaulting to 0.25), then the object is considered detected.

For large objects spread over several cells (of the S x S grid size), it may be that, upon inference, several of these cells predict a bounding box for this object. A post-processing method is therefore needed to prune and retain the best predictions on an image.

The Non-max Suppression (NMS) technique is applied for each class. The box with the highest confidence score is selected. The IoU between it and the others is calculated.

All boxes with an IoU above a predefined threshold with this box are eliminated. We repeat the process on the remaining boxes until there are none left. Finally, among bounding box clusters (in the sense of IoU), those with the highest confidence score are selected.

C. Architecture

Computer vision has witnessed tremendous advancements in recent years, enabling machines to perceive and understand the visual world with remarkable precision.

One of the key developments in computer vision is the YOLO series, which has consistently improved its performance with each subsequent update. From YOLOv2 to the latest iteration, YOLOv8, significant enhancements have been made to both the loss function and network structure to improve overall performance. These improvements have made YOLOv8 a highly efficient and accurate object detection algorithm. ## YOLOv8: A Single-Stage Algorithm YOLOv8 is a single-stage algorithm that utilizes a complex Convolutional Neural Network architecture.[5]

This architecture represents a significant improvement over previous versions, such as YOLO Redmon et al and YOLOv2 Redmon and Farhadi. The YOLOv8 architecture is primarily based on Darknet-53.

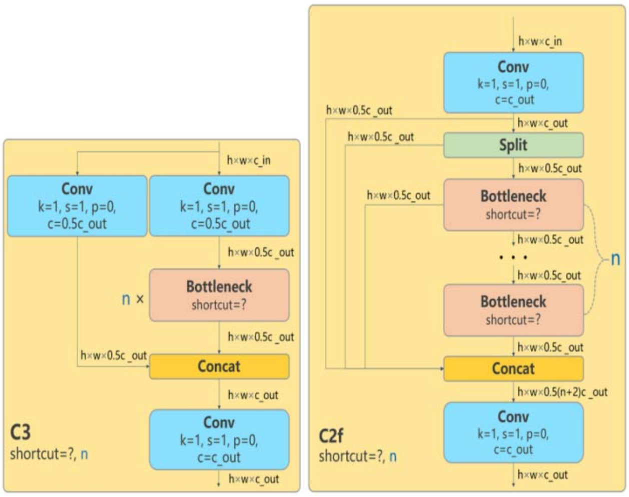

Fig. 2: YOLOV8 architecture

The YOLOv8 architecture boasts several key features that contribute to its superior performance:

1. Improved Loss Function:

YOLOv8 incorporates an improved loss function compared to previous versions, resulting in better object detection accuracy.

2. Complex CNN:

YOLOv8 utilizes a complex Convolutional Neural Network architecture, which allows for more accurate and efficient object detection.

3. Real-time Performance

YOLOv8's unique single inference mechanism enables it to achieve high real-time performance, making it an ideal choice for applications that require fast and accurate object detection.

4. Continuous Improvement:

The YOLO series, including YOLOv8, has undergone continuous development to enhance its overall performance. These improvements include modifying the loss function and optimizing the network structure, resulting in increased accuracy and efficiency.

5. Faster and More Capable*:*
YOLOv8 has shown significant improvements in speed and capability compared to its predecessors. YOLOv8 is a faster and more capable candidate for real-time object detection, surpassing previous versions such as YOLOv2 and Yv3.[8]

Results and discussion

The first part of the work consists in testing the pertained yolov8 model on images, videos and in real time. The object chosen for image reconnaissance is a knife, as it is dangerous thus, it would be a ways of preventing danger and insecurity.

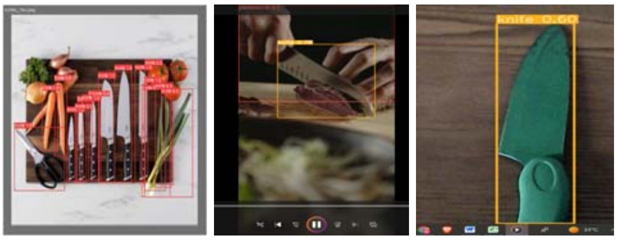

Fig. 3: Testing yolov8 on an image, video and in real time

The second part is training a custom YOLOv8 model to recognize a single class (knife) using data from Google.

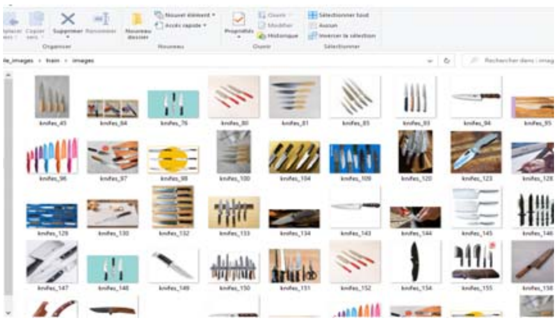

Fig. 4: The sata used for training YOLOV8

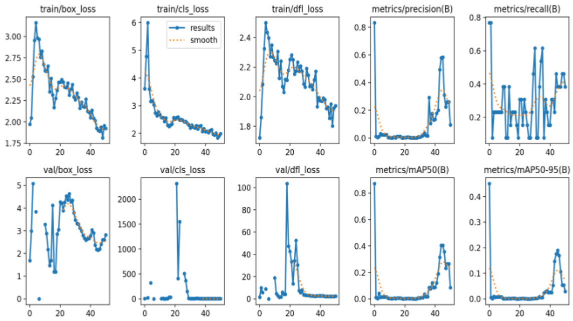

Fig. 5: Training yolov8 on custom dataset

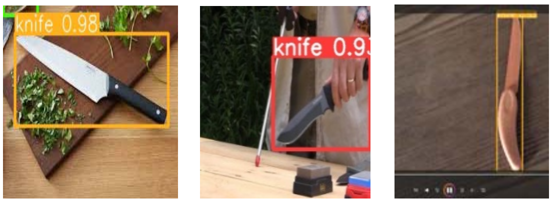

Fig. 6: TestingYolov8 custom on image, video and in real time

The last part of the work is to train the yolov8 model using data taken from the everyday environment.
Quite simply, photos of kitchen knives taken:

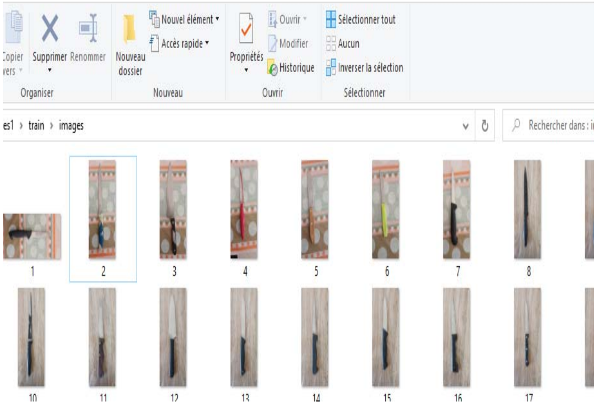

Fig. 7: The Data used for training YOLOV8(Kitchen knives)

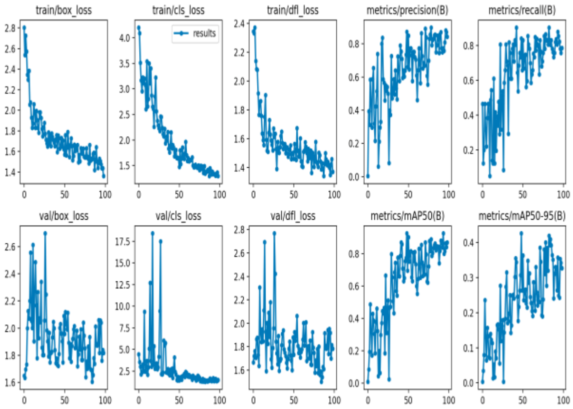

Fig. 8: Training yolov8 on custom dataset (Kitchen knives)

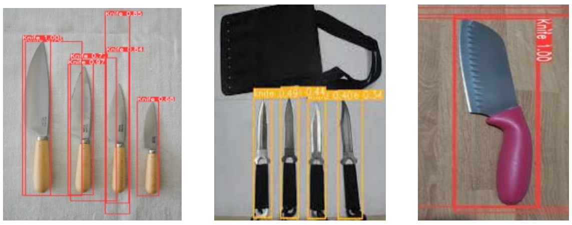

Fig. 9: TestingYolov8 custom on image, video and in real time

A. Confusion matrix

Yolov8 pretrained model:

Images

	Predicted	
Actual	19	2
	1	17
Acurancy		

Video

	Predicted	
Actual	4	1
	2	5
Acurancy		

Real Time

	Predicted	
Actual	2	6
	2	1
Acurancy		

Yolov8: custom Google data set

Images

	Predicted	
Actual	17	2
	5	13
Acurancy		

Video

	Predicted	
Actual	4	3
	2	4
Acurancy		

Real Time

	Predicted	
Actual	2	6
	2	1
Acurancy		

Yolov8: custom Google data set

Images

	Predicted	
Actual	18	3
	1	15
Acurancy		

Video

	Predicted	
Actual	3	1
	1	4
Acurancy		

Real Time

	Predicted	
Actual	5	1
	0	1
Acurancy		

Conclusion

The usefulness of object recognition is often greater when applied in real time.

The work presented is a study of the Yolov8 algorithm, which is the latest version of YOLO.

After a general presentation and explanation of YOLOV8 and the evolution of yolov over the years, a comparative study of the 3 following data phases is presented:

- We tested the pre-trained yolov8 model on knife images, on video and on images, video and in real time and the results were reliable and robste on HD images and video.

The accuracies of the three trials successively were 0.92, 0.75 and 0.27.

We can therefore see that in real time and using a simple webcam the efficiency of yolov8 is very reduced.

-we then trained yolov8 with google data, sharp images of knives and tested the model again on images, videos and in real time.

The accuracy is alternately 0.81, 0.61 and 0.62.

We can clearly see that there is a performance improvement in the real-time test.

-Finally, we trained the yolov8 model with photos taken by cameras, and the results changed dramatically: 0.63 on images, 0.61 on videos and 0.85 in real time.

As already mentioned, the usefulness of object recognition, especially dangerous objects such as "knife" in our case, is essentially in real time.

In perceptive, more variation of data will be practical as for expemple vary between images of google and photos taken by cameras.

***Authors**: Maha Mokrani ph.d student (email: mokranimaha@gmail.com, phone: +216 25049981) ATSSEE FST, Université de Tunis El Manar*
Pr:Zied Hajaiej (email: Zied.hajaiej@fsb.ucar.tn, phone: +216 23 044 545) ATSSEE, FST, Université de Tunis El Manar

REFERENCES

[1] Juan. R. Terven, Diana M Cordova, A COMPREHENSIVE REVIEW OF YOLO: FROM YOLOV1 AND BEYOND, UNDER REVIEW IN ACM COMPUTING SURVEYS

[2] Yiting Li and. Al, A Modified YOLOv8 Detection Network for UAV Aerial Image Recognition, Academic Editor: Anastasios Dimou

[3] Y. Zhou, W. Zhu, Y. He and Y. Li, "YOLOv8-based Spatial Target Part Recognition," 2023 IEEE 3rd International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA), Chongqing, China, 2023, pp. 1684-1687, doi: 10.1109/ICIBA56860.2023.10165260.

[4] R. Bawankule, V. Gaikwad, I. Kulkarni, S. Kulkarni, A. Jadhav and N. Ranjan, "Visual Detection of Waste using YOLOv8," 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS), Coimbatore, India, 2023, pp. 869-873, doi: 10.1109/ICSCSS57650.2023.10169688.

[5] M. Karthi, V. Muthulakshmi, R. Priscilla, P. Praveen and K. Vanisri, "Evolution of YOLO-V5 Algorithm for Object Detection: Automated Detection of Library Books and Performace validation of Dataset," 2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), Chennai, India, 2021, pp. 1-6, doi: 10.1109/ICSES52305.2021.9633834.

[6] T. S. Gunawan, I. M. M. Ismail, M. Kartiwi and N. Ismail, "Performance Comparison of Various YOLO Architectures on Object Detection of UAV Images," 2022 IEEE 8th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA), Melaka, Malaysia, 2022, pp. 257-261, doi: 10.1109/ICSIMA55652.2022.9928938.

[7] G. Yasmine, G. Maha and M. Hicham, "Overview of single-stage object detection models: from Yolov1 to Yolov7," 2023 International Wireless Communications and Mobile Computing (IWCMC), Marrakesh, Morocco, 2023, pp. 1579-1584, doi: 10.1109/IWCMC58020.2023.10182423.

[8] P. Chen, Y. Shi, Q. Zheng and Q. Wu, "State-of-the-art of Object Detection Model Based on YOLO," 2020 International Conference on Computer Network, Electronic and Automation (ICCNEA), Xi'an, China, 2020, pp. 101-105, doi: 10.1109/ICCNEA50255.2020.00030.