Multi Moving Objects Detection in Video Using Pre-trained Deep Convolutional Neural Networks

Abolfazl Ansaripour
dept. of Artificial Intelligence University of Isfahan
University of Isfahan
Isfahan, Iran
Email: Ansar2010 [AT] gmail.com

Hosein Mahvash Mohamadi
dept. of Artificial Intelligence University of Isfahan
University of Isfahan
Isfahan, Iran
Email: h.mahvash [AT] eng.ui.ac.ir

Abstract— **Abstract- Nowadays object tracking is a critical concern in the field of machine vision. With the advent of powerful computers, affordable cameras, and growing demand for automatic video analysis, researchers have shown significant interest in object tracking. Various methods have been proposed for tracking objects in machine vision, but a key challenge remains: ensuring the robustness of tracking algorithms across consecutive video frames. In recent years, deep neural networks have emerged as a promising approach for accurate position estimation. In this study, we propose an enhanced method that combines deep convolutional neural networks with established techniques like K-means clustering. Our approach addresses challenges such as object disappearances and severe displacements. The selection of deep neural networks is motivated by their compatibility with target identification in video sequences, and achieving a remarkably low error rate in tracking validates our claim.**

Keywords: Object Tracking; Video processing; Deep neural network; K-mean clustering.

I. INTRODUCTION (HEADING 1)

The In recent decades, scientific advancements have significantly impacted human life. The integration of technology into personal and societal contexts has transformed our daily experiences. Concurrently, the exponential growth in hardware processing power has fueled diverse applications in multimedia. Among these challenges, tracking objects within video sequences has garnered attention due to its wide-ranging applications across various fields.

The ubiquitous use of cameras in today's societies, particularly in industrialized countries, is undeniable. Enhancing camera quality and features has become a crucial factor in improving their performance. Within the realm of science, machine vision studies these cases extensively. Among its critical topics, tracing moving objects within images stands out due to its real-world implications. A moving object tracking system finds applications in various contexts, including smart robots. These robots, operating in dynamic environments, must interact with other objects—humans, environmental elements, or fellow robots. By incorporating motion estimation to prevent collisions, robots can navigate freely in ever-changing surroundings. Additionally, monitoring and surveillance systems heavily rely on target tracking. In such systems, one or more cameras continuously monitor specific locations—be it a shopping mall, airport, museum, military base, or any security-sensitive area. Depending on the application, augmenting the system with features like face recognition or license plate recognition enhances monitoring accuracy. Scientific advancements have profoundly impacted human life, bridging the gap between humans and machines through technology.

In the realm of machine vision science, a primary objective is to enhance the intelligence of cameras for applications in surveillance, commerce, military, and related fields. Extensive research has been dedicated to developing novel intelligent methods and refining existing ones. Much of this research centers around the identification and tracking of targets. The overarching goal is to streamline computations while improving the accuracy of detection and tracking processes. Target detection involves identifying image regions that may correspond to the desired target. Tracking systems are categorized based on their specific applications, with cameras and targets serving as pivotal components. Just as these components shape the nature of tracking systems, they significantly influence the choice of methods employed. System configurations vary based on factors such as the number, type, and environmental conditions of cameras and targets.

Changes in the structure of tracking systems can lead to major changes in the tracking methods, these methods have many commonalities in many respects and principles. The main difference between these methods is in the way the general steps are performed. In this paper, the algorithms used in goal tracking systems are divided into two main categories based on the use of predictability. This means that as the target area in each frame is specified and the next frame arrives, the area of the next frame that most closely resembles that area is considered the target area in the next frame. In other words, if the target position was available in frame k, an area would be considered the target area in that frame. Arriving at frame k + 1, while searching around the previous position of the target, we try to find an area in frame k + 1 that most closely resembles the target area in frame k. The criterion of similarity is generally considered to be the minimum error for mean of

the squares. In these methods, information about how the target moves is not used much. In other words, according to the movement direction of the target, no prediction is made about its position in the next frame. These algorithms are generally very fast due to their low volume of operations. But because they usually use the general characteristics of the target area to create matching, in some situations they have relatively large errors. Also, due to the general characteristics used in these methods, they have very little resistance to changes in target conditions. However, due to the mentioned advantages, the methods of this category are still very popular. Algorithms such as Mean Shift and CAM Shift are the most popular examples of this method.

In the second category, algorithms with prediction properties, the target position in frame number k is used to make a prediction of the target position in frame k + 1. In these methods, the location of the target in the next frame is specified and the search for the best corresponding area of the target is done in a much smaller area. Of course, it should be noted that calculating the prediction of the target position in the next frame increases the calculations in this category of methods. Several methods have been developed to predict the position of the target in the next frame, most of these methods are based on Bayesian family filters. Algorithms based on Bayesian filters generally have similar performance. Their general function is that at moment k the position of the target in the frame k is known. However, the system can predict the position of the target at k + 1 depending on the type of target movement. This prediction is based on the type of movement and taking into account different assumptions for the tracking system model. Reaching frame number k + 1, the system corrects its previous prediction value by obtaining the actual position of the target.

These methods are efficient. However, from a practical point of view, they are not very popular. The main reason for this is the high volume of operations required, which causes limitations in the use of these methods. Tracking algorithms based on Kalman filter or particle filter are well-known examples of this type of algorithm.

So far, various methods have been introduced to identify the objects in the image and understand their meaning. These methods can be divided into three general categories:

In the first category, a set of images of different states of the object is created, which are called patterns. They then compare the received image with all of these patterns. If the desired match is achieved between the received image and each of the patterns, the presence of that object in the image can be reported. This method is called pattern matching. In order to identify the signposts, White and Arocleo have used the pattern matching method by calculating the Hasdorf distance between the desired parts in the image and the defined patterns [1]. They have examined the average similarity of the windows with the pattern by considering equal size windows with the dimensions of the pattern on the image and by the square error method. [2]

In the second category, a set of patterns is created for the object, which are called training patterns. In these methods, with the help of these models, machine learning algorithms [3] or neural networks are taught. After a proper training with sufficient patterns, the algorithm used can identify the object in the image. This method is called identification through training. The algorithm training process takes place in the laboratory and then there is no need for a set of training models. One of the most common machine learning algorithms is the support vector machine.

The application of this algorithm is in classifying samples. This algorithm has been used to detect traffic signs [4] and to identify them [5 - 6]. Nearest Neighbor Algorithms [7] and Decision Trees [8] are also well-known machine learning algorithms used to identify objects. The neural network is also a powerful learning algorithm that is very useful for identifying objects [9 -10].

The third category examines the properties of the object within the image. Attributes are a collection of information extracted from an object image that represent that object and enable analysis for the recognition algorithm. In this method, how to select the features and the recognition algorithm are very important. Oroklohe et al have used the SURF explicit stable properties algorithm to identify objects. The SURF method consists of two parts: diagnostic and descriptor. In the descriptor section, the desired properties are assigned to specific areas of the image, such as corners, dots, etc., and in the descriptor section, the neighbors of the feature points are described by the feature vector [11].

In [12], a solution for identifying human movement using light flow in video is presented, which is able to predict the next movement using different three-dimensional image of human. The proposed method is based on SpyNet.

Also [13] presents an integrated system that combines computer vision and deep learning techniques to enhance object tracking in videos. This approach leverages the YOLOv8 architecture for accurate object localization and predict bounding box locations in video frames using deep learning, allowing to extract precise object positions. Blurring and optical flow analysis aid in precise object tracking, while optical flow maps the object's movement across frames, enabling accurate trajectory tracing. This comprehensive approach ensures consistent object identification throughout the validation video. and results are based on the DFL Soccer ball detection dataset, where this method shows promising results in real-time football tracking during matches. The proposed system has applications in surveillance, autonomous navigation, and more.

In this paper, in the second part, the basics of the research are presented to prepare the necessary introduction for present the proposed method. The third part of this paper is dedicated to presenting the proposed method and in the fourth part, the simulation is performed along with the results related to the proposed method. Finally, in the fifth section, the summary and conclusion of the proposed method are presented.

II. BASICS

A. Convolutional neural network

In this article, the basis of the proposed method is based on the use of deep neural networks, so in this section, we will dedicate to presenting the conventional structures of deep neural networks and its generalities.

Today, Convolutional neural networks are very applicable and similar to scientists. These types of networks consist of neurons with learnable weights and biases. Each neuron receives a number of inputs and then calculates the product of the weights in the inputs and finally presents a result using a one-line nonlinear conversion (activation) function. The entire network also provides a derivative score function, with the raw pixels of the input image on one side and the scores for each category on the other. These types of networks still have a cost function such as SVM, Softmax (in the last layer) all related, and all the points made about conventional neural networks are true here as well.

After presenting the basics of the research, the proposed method will be presented in the next section.

III. PROPOSED METHOD

The proposed method can be summarized in two separate sections: identifying object in each frame and tracking them. The details of each section will be explained in detail below.

A. Recognize objects

Object recognition is a widely used technology in the field of machine vision and image processing, which is examined by identifying examples of specific objects of a particular category (such as humans, buildings and cars, etc.) in digital images and videos.

In object recognition the image will be analyze and finally the location of the objects is defined and their class or category is also specified. For an overview of the training images reveals that sometimes the number of data in some classes in the data set is much higher than in other classes. Fig. 1 shows the distribution of 43 alternating classes in the Open Image dataset.

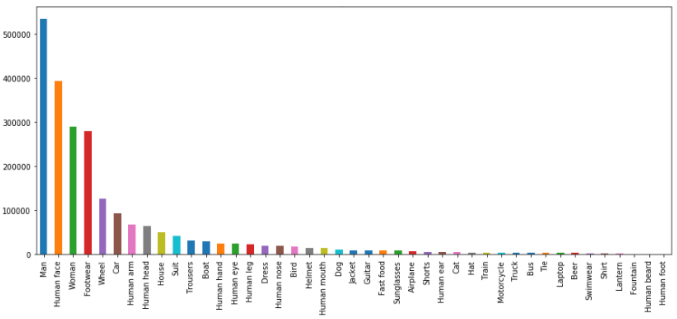

Figure 1. Distribution of samples in the Open Image database.

According to the figure distribution of samples in different classes is unequal; The problem that needs to be solved somehow for optimal object recognition system training. Only the data in the 43 classes in the Open Image dataset are used to teach the object recognition system (there are a total of 300,000 images in these 43 classes).

To balance the educational data frequency of each class, only 400 images from each class are randomly selected and placed in the educational set, and it should be noted that a total of 17,200 images have been used to implement the object recognition system in this article.

During the implementation of this paper, various algorithms were considered to implement the object recognition system, and finally, the YOLO algorithm was considered as the main algorithm. The reason of choosing the YOLO algorithm is its high speed and computation power, as well as the availability of many training resources when implementing this algorithm. Due to computational and time constraints, the following decisions were made to design and implement the neural network (for object recognition applications). A YOLO V3 model that has already been trained to identify specific objects is used (a pre-trained deep learning model). In the first version of this network, after the convolution layers, the features are linearized and delivered to all connected layers, and finally the predictions are done, while in the next versions, instead of creating a tensor to these large dimensions, a convolution kernel with The dimensions of K (C + 1 + 4) are used for characteristics of the last layer of convolution so that its receptive fields are located on each of the N * N areas of the input image and the prediction operation is performed. The YOLO V3 uses transitional learning, and the last convolutional layer of the YOLO model is re-trained so that the object recognition system can detect objects that it has not encountered before, or in other words test data such as guitar, house, man, Female, bird and other items, recognize.

B. Specifications of YOLO model for object recognition

The YOLO algorithm accepts inputs with the following specifications in order to be able to detect objects in the image:

- The dimensions of the input images in YOLO network are equal to 416 × 416.
- The number of classes for network training is equal to 43 classes. The number of classes is a parameter needed to define the output dimensions of the YOLO network.
- Anchor Box parameters include the number of enclosing frames and their maximum and minimum dimensions.
- Threshold for IoU [1] criteria and reliability: This threshold specifies which of the enclosing boxes should be selected as the enclosing box of the objects in the image (choice between enclosing boxes).

C. YOLO architecture

The YOLO network architecture is shown in the Fig. 2. This network consists of 23 layers of convolution; Each of

[1] Intersection over Union

these layers has its own batch normalization unit, attenuated relay activation function, and its own maximum pulling unit.

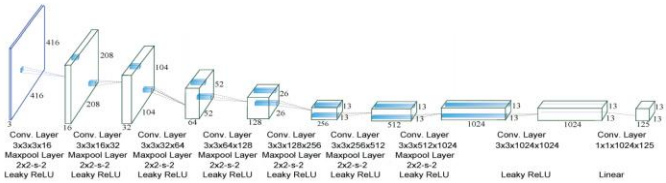

Figure 2. YOLO network architecture.

The purpose of the layers defined in this architecture is to extract several important features of digital images in order to identify the different objects in the image and classify them into corresponding classes. In order to detect objects in the image, the YOLO algorithm divides the image into a grid consisting of 19×19 cells and defines five enclosing frames with different dimensions for each cell. The YOLO network then tries to identify the class of objects in the grade cells; In other words, the probability of each of the identified objects belonging (within the enclosing boxes of each cell) to the classes in the data set is calculated.

Each enclosing box has different dimensions and shapes relative to each other and is essentially designed to identify different objects (with different shapes and sizes) in each of the grade cells. The output of the matrix YOLO algorithm is as follows; For each of the defined enclosure boxes (in each of the grade cells), a matrix similar to the following figure will be generated.

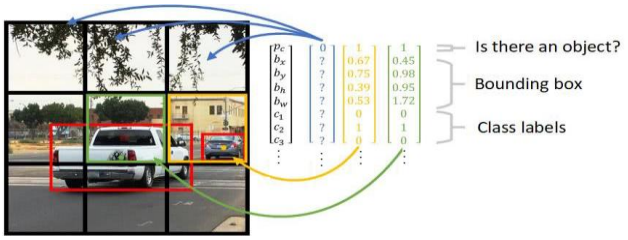

Figure 3. YOLO algorithm output.

It should be noted that in Fig. 3, Bx and By correspond to the position of the enclosing box, and Bw and Bh indicate its width and height. Since the YOLO network is taught using the images of 43 classes, the dimensions of the output matrix are calculated as follows. In these outputs

This matrix provides very important information such as the probability of an object being observed in each of the enclosing boxes and the probability that the object identified in the enclosing card belongs to each of the predefined classes. To set non-object boxes (boxes in which there is no specific object), boxes in which the detected object is not categorized in any class, or boxes in which the detected object overlaps with other boxes to be filtered at two thresholds The following is used:

- IoU threshold: Used to filter boxes in which a single object is detected.
- Reliability threshold: Used to filter boxes, they are very unlikely to belong to different classes.

Grid size: In the pre-trained YOLO model, the dimensions of the grid cells are 13× 13 ×13. In this study, the dimensions of the grid cells were changed to 19 × 19 to retrain the YOYO model.

1) Customize model

In order to be able to train the YOYO model on new image data, it is necessary to reset the weight parameters of the last layer of convolution; This allows the system to properly categorize images belonging to specific and new classes.

2) Cost or loss function

In matters related to the recognition of objects in the image, the goal is to correctly identify the class of objects, with the highest probability or reliability. The cost function for an object recognition system consists of three components:

a) Classification losses

If an object is detected in the image, this component calculates the conditional probability error square of the class. Therefore, the loss function only imposes a penalty for a classification error if an object is present in a grade cell.

b) Localization losses

This component calculates the square error associated with the size and location of the enclosing frame responsible for recognizing the object relative to "the size and location of the enclosing frame in the correct base responses" if the enclosing knives are responsible for recognizing the object in the image. An adjustment parameter is used to determine the penalty for errors resulting from the spatial coordinates of the enclosing knives.

c) Confidence loss

This component calculates the squared error square of the enclosing box. Since not all enclosing frames defined define the object in the image, the loss calculation equation in this component is divided into two parts; The first part is for the enclosing boxes that are responsible for recognizing the objects in the image, and the second part is for the other enclosing frames.

The main advantage of the YOLO model over similar object recognition models is that it calculates errors that can be easily optimized by well-known optimization functions such as random reduction gradient, random reduction gradient with momentum parameter and Adam optimizer.

To summarize, we can finally show how the object in the image is detected by the YOLO algorithm as follows: It is assumed that K objects are possible. The grid is designed to ultimately predict the possibility of objects in those areas - the handle of the object, as well as the coordinates of the frame around the object - for all areas.

Given that the box around each object has 4 coordinates and also the desired category of the object is a C vector and there is also a number for the probability of the object, then for each object a tensor (C + 1 + 4) is required.

According to this fact, it is assumed that in each image there is N in N area and each area has a maximum probability of S object presence and each object also needs a tensor with dimensions (C + 1 + 4), so at the end of that there is a tensor network that predicts the frame of objects and their probability and their category. It should be noted that the innovation considered in this paper is the use of the K-Mean clustering algorithm to determine the best tensor for the enclosing frame. The choice of the K-Mean algorithm has been made due to its excellent performance in machine vision processes and pattern recognition. It should be noted that experimentally the value of 9 for K has been associated with the best results.

D. Target tracking

To implement the tracking process in consecutive frames, the position of the enclosure box center is calculated and then the target position in the next frame will be calculated. Finally, having the position of the centers of the cadres and the correct position of the base for each target in successive frames, the percentage of accuracy or error of the tracking process is calculated.

To evaluate the output, first the object detection process and then the target tracking process will be examined separately. Several criteria have been proposed to evaluate the object recognition process. One of the most important of these is IOU criterion.

E. IOU criterion

This criterion measures the degree of overlap between two regions; This value is equal to the area resulting from the overlap of these two areas, divided by the area resulting from the union of these two areas. This criterion basically shows the quality of the predictions produced by the object recognition system to the correct answers of the basis and compares them with each other.

F. Objects recognition

As mentioned in the previous section, the first step in implementing the proposed method in goal tracking is to implement the goal recognition process in consecutive frames. In this section, it describes its implementation and the result of its implementation on several examples of challenging frames. We present the motivation.

1) Training data

In this study, we employed the Open Image dataset for training our convolutional neural network. Specifically, we utilized data from 43 distinct classes within the dataset, resulting in a total of 300,000 images. To address potential data imbalance, we randomly selected 400 images from each class, ensuring a balanced representation. Ultimately, our object recognition system was implemented using a total of 17,200 images. Figure 4 provides illustrative examples from our dataset.

Figure 4. An example of the images used to teach neural networks using the open dataset database.

2) Convolutional network

In the object detection process, the input images are divided into s×s pixel sections at the beginning of the work using the desired convolutional network. In each section k enclosing box is considered that the probability of the presence of the object in each enclosing box is calculated from the Eq. 1.

$$\Pr(\text{Object}) * \text{IoU}_{\text{pred}}^{\text{truth}}$$

In the above phrase, Pr can be rewritten according to the principle of conditional probabilities and the distribution of probabilities as Eq. 2.

$$\Pr(\text{Class}_i \,|\text{Object}) * \Pr(\text{Object})$$

Term $\text{IoU}_{\text{pred}}^{\text{truth}}$ also indicates the degree of overlap between the two areas, which was examined in detail in the previous section. Using the above relations, the degree of confidence between the staff is considered and the correct criterion is introduced as Eq. 3.

$$\Pr(\text{Class}_i \,|\text{Object}) * \Pr(\text{Object}) * \text{IoU}_{\text{pred}}^{\text{truth}} = \Pr(\text{Class}_i) * \text{IoU}_{\text{pred}}^{\text{truth}}$$

It should be noted that the input sample of the first convolution layer is specified by xl, the weight of the convolution kernel of that layer is specified by w^l and its bias is denoted by b^l, and the intermediate variable or output of that layer is calculated from the Eq. 4.

$$\boldsymbol{y}^l = \boldsymbol{x}^{l-1} * \boldsymbol{w}^l + \boldsymbol{b}^l \tag{4}$$

If the operator f is considered an activation function, in the error propagation process, the relationship between each layer and the previous layer will be specified as Eq. 5.

$$\boldsymbol{x}^l = f\left(\boldsymbol{y}^l\right) = f\left(\boldsymbol{x}^{l-1} * \boldsymbol{w}^l + \boldsymbol{b}^l\right)$$

It should be noted in Eq.5, l is the layer number. Operator L represents the loss function, and in the gradient error post-propagation process, the loss function is calculated from the Eq. 6.

$$\boldsymbol{\delta}^{l-1}=\frac{\partial \boldsymbol{L}}{\partial \boldsymbol{y}^{l-1}}=\frac{\partial \boldsymbol{L}}{\partial \boldsymbol{y}^{l}}\cdot\frac{\partial \boldsymbol{y}^{l}}{\partial \boldsymbol{y}^{l-1}}=\boldsymbol{\delta}^{l}*\mathrm{rot180}\left(\boldsymbol{w}^{l}\right)\odot f'(\boldsymbol{x}^{l-2}*\boldsymbol{w}^{l-1}+\boldsymbol{b}^{l-1}) \tag{6}$$

In the Eq. 6, the rot180 operator for rotating 180 degrees is the weight component counterclockwise and Θ is the Hadamard multiplier. As the gradient moves between the layers, the weight component of the layer gradually decreases, and this occurs due to the multiplication of the derivative of the activation function in the gradient. For example, the derivative of the sigmoid activation function would be as Eq. 7.

$$\left|f'(\boldsymbol{y}^{l-1})_{\mathrm{Sigmoid}}\right|\le 1/4 \tag{7}$$

This value is always less than one and during the post-propagation process the error in the network tends to very low values and the gradient fading process occurs which will eventually lead to a significant decrease in the accuracy percentage.

Figure 5 shows the structure of the convolutional neural network under training.

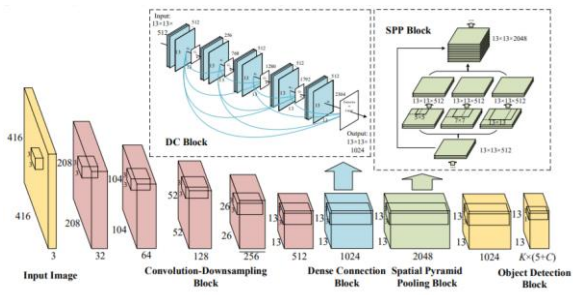

Figure 5. CNN neural network Structured

In the above structure, two components of dense connection block and spatial pyramidal polishing block can be seen, which are described below.

G. *Dense connection block*

As mentioned in the previous section, one of the network problems used is the gradient fading process. In this paper, dense connection blocks have been used to solve this problem. In the figure below, this block can be seen in better detail.

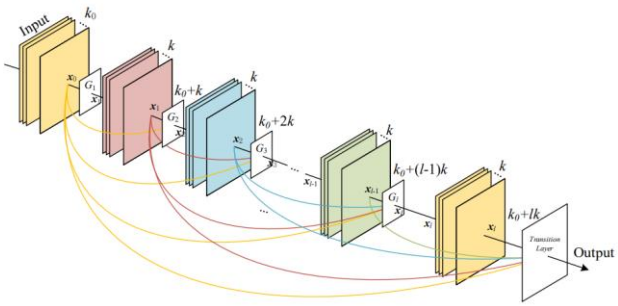

Figure 6. Dense connection block

In this block, this problem is solved by using the feature layer of the previous layers, and also the input of the 1st layer is calculated as Eq. 8.

$$\boldsymbol{x}^{l}=f\left(\boldsymbol{y}^{l}\right)=f\left(\left[\boldsymbol{x}^{0},\boldsymbol{x}^{1},\cdots,\boldsymbol{x}^{l-1}\right]*\boldsymbol{w}^{l}+\boldsymbol{b}^{l}\right) \tag{8}$$

In the error propagation process, the loss function gradient will be as Eq. 9.

$$\boldsymbol{\delta}^{l-1}=\boldsymbol{\delta}^{l}*\mathrm{rot180}\left(\boldsymbol{w}^{l}\right)\odot f'(\left[\boldsymbol{x}^{0},\boldsymbol{x}^{1},\cdots,\boldsymbol{x}^{l-2}\right]*\boldsymbol{w}^{l-1}+\boldsymbol{b}^{l-1}) \tag{9}$$

In Eq. 9, the part $f'(\left[\boldsymbol{x}^{0},\boldsymbol{x}^{1},\cdots,\boldsymbol{x}^{l-2}\right]*\boldsymbol{w}^{l-1}+\boldsymbol{b}^{l-1})$ will always include the input of layer x^0 and the output of the map of the properties of the previous layers, and finally, $f'(\boldsymbol{x}^{l-2}*\boldsymbol{w}^{l-1}+\boldsymbol{b}^{l-1})$ in comparison with the expression, it solves the problem of gradient vanishing.

H. *Spatial pyramid blocking*

Another thing that can be seen in this structure is that in the conventional structure it focuses only on the global properties of the multi-scale convention layers, while the fusion ignores the local properties of the multi-scale localities in the same annular layer. In this paper, a new spatial pyramid block is used to extract and assemble multi-scale local features, and finally these global and local multi-scale features are used to improve object recognition accuracy. This structure can be seen in the Fig. 8.

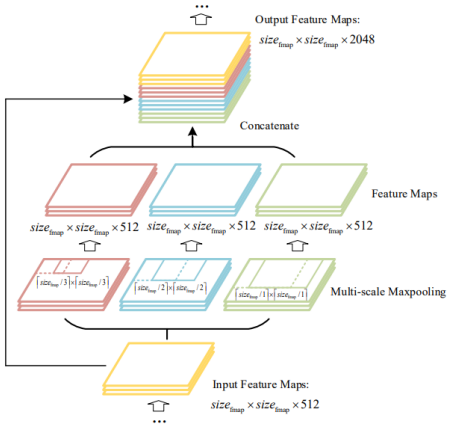

Figure 7. Spatial pyramid polishing block

In Fig. 7, three maximum polishing layers are considered between the dense block and the object recognition layer. In the Fig. 7, a convolution layer is first used to reduce the input feature map from 1024 to 512. Then, using the polishing layer, the dimensions of the feature map are changed to the dimensions shown in the image. Finally, the dimensions of the feature map at the output change to 2048.

In the Table 1, the network components can be seen for an image with dimensions of 3 × 416 × 416, the length and width of the image will be 416 pixels, respectively, and the number of color channels will be three (red, green and blue).

As mentioned in the previous chapter, each enclosing box is as Eq. 10.

$$b=[b_x\ b_y\ b_w\ b_h\ b_c]^t$$

TABLE I. COMPONENTS OF DEEP CONVOLUTIONAL NEURAL NETWORK USED

Layers	Parameters		Output	Layers	Parameters		Output
	Filters	Size / Stride			Filters	Size / Stride	
Conv 1	32	3×3/ 1	416×416×32	DC Block	1024	3×3 / 1 ×4	13×13×2304
Maxpool 1		2×2 / 2	208×208×32	Conv 14-21	256 or 512	1×1 / 1	
Conv 2	64	3×3 / 1	208×208×64	Conv 22	1024	3×3 / 1	13×13×1024
Maxpool 2		2×2 / 2	104×104×64	Conv 23	512	1×1 / 1	13×13×512
Conv 3	128	3×3 / 1	104×104×128	SPP Block Maxpool 6-8		5×5 / 1 7×7 / 1 13×13 / 1 Concat	13×13×2048
Conv 4	64	1×1 / 1	104×104×64				
Conv 5	128	3×3 / 1	104×104×128				
Maxpool 3		2×2 / 2	52×52×128	Conv 26	512	1×1 / 1	13×13×512
Conv 6	256	3×3 / 1	52×52×256	Conv 27	1024	3×3 / 1	13×13×1024
Conv 7	128	1×1 / 1	52×52×128	Reorg Conv13		/ 2	13×13×256
Conv 8	256	3×3 / 1	52×52×256	Concat -1, -2			13×13×1280
Maxpool 4		2×2 / 2	26×26×256	Conv 30	1024	3×3 / 1	13×13×1024
Conv 9-12	512	3×3 / 1 1×1 / 1 ×2		Conv 31	Conv31	K*5+C	1×1 / 1	13×13×(K*5+C)
Conv 13	512	3×3 / 1	26×26×512	Detection			
Maxpool 5		2×2 / 2	13×13×512				

Where b_x and by refer to the center of the enclosing box, b_w and b_h refer to the width and height of the enclosing box, and finally b_c refers to the level of confidence in the classification of the enclosed object in the box.

The Fig. 8 show the result of running a pre-trained network for several test video sequences that have a benchmark. In this sequence, several moving people or several planes flying at the same time are detected and followed with good accuracy.

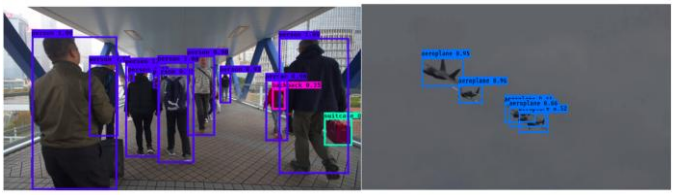

Figure 8. The result of performing the object detection

After completing this step, the next step is dedicated to the tracking process.

I. Target tracking

After obtaining the target specifications, it is time to track based on the findings of the previous step.

To implement this step, three criteria b_x, b_y and b_c are used. The procedure is such that in order not to lose the subject in consecutive frames in each frame, a list of each enclosing box consisting of the above three components is formed, and finally, with the help of the performed processing, the position of a specific target in the next frame is formed. Extracted and finally with the help of that path the subject is extracted in consecutive frames. It should be noted that to determine a specific subject in the case that the goals are close to each other, first the Euclidean distance criterion is used according to the Eq.11.

$$dis = \sqrt{(b_{x1} - b_{x2})^2 + (b_{y1} - b_{y2})^2} \quad (10)$$

In the above statement, b_x^1 and b_y^1 correspond to the position of the center of the enclosing frame in the reference frame, and b_x^2 and b_y^2 correspond to the position of the center of the enclosing frame in the next frame. If the distance difference for two or more goals is less than the threshold, the third measurement criterion, b_c, will determine the selected goal in the next frame. Thus, the criterion for measuring the smaller value of the difference will be bc_1 and bc_2, and the enclosing box for which the value of this difference is less in the future frame will be selected as the target position in the future frame. The Fig. 9 shows the simultaneous target.

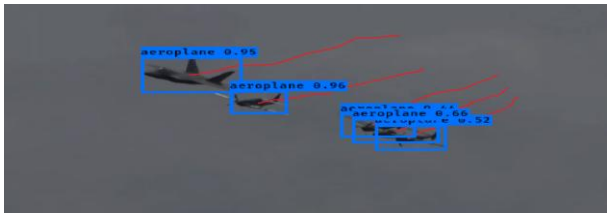

Figure 9. Object detection through successive frames of an example video sequence

IV. RESULTS

To assess the proposed method, we compute the difference between the center position of the bounding frame obtained by our approach and the predefined values for each frame and target. Additionally, we evaluate the method's speed performance by calculating the target finding time based on the number of subjects in the image for online applications. Table II presents the average error percentage across all frames.

TABLE II. MEAN ERROR FOR ALL TARGETS IN ALL FRAMES

Method	**Test Video Sequence**			
	scarfing	***Walking***	***Street***	***Airplane***
Proposed	**%3.17**	**%2.45**	**%6.27**	**%2.03**
SPyNet [12]	**%7.18**	**%6.03**	**%12 . 11**	**%3.27**

Table III also shows the average tracking time. This criterion is normalized according to the number of targets and the average is reported on the whole frames.

TABLE III. AVERAGE TRACKING TIME FOR ALL TARGETS IN ALL FRAMES

Method	**Test Video Sequence**			
	scarfing	***Walking***	***Street***	***Airplane***
Proposed	**1.24**	**0.86**	**1.15**	**1.11**
SPyNet [12]	**2.17**	**1.98**	**2.02**	**1.17**

V. CONCLUSION

In this study, we introduced an efficient method for object interception in video sequences using deep neural networks. Leveraging the advantages of deep learning, our proposed approach improves accuracy and minimizes interception errors. The method comprises two main components: object recognition, where all objects in each video frame are identified, and error reduction, which aims to minimize discrepancies. To achieve this, we employed a deep convolutional neural network (YOLO) trained on a valid dataset. The extracted output from this stage directly influences the final results. Additionally, the output tracking component plays a crucial role. The matrix comprises object coordinates (center, width, and height) along with confidence levels for subsequent steps. In the second phase, the algorithm processes the test video sequence, extracting a matrix of numerical values corresponding to each object in every frame. In all cases, our proposed method generates a corresponding matrix for each object in every frame. This matrix is evaluated by comparing the center distance obtained using our method with the expected values from the proposed algorithm. The resulting error and overall performance validate the effectiveness of our approach. The second evaluation criterion assesses the detection time for each target across all frames, represented as the average result time. This metric directly reflects the method's suitability for online applications. Analyzing the results presented earlier, we conclude that our proposed method performs well in tracking moving targets, achieving a low error rate of 2.03% and an efficient time of 0.86 seconds compared to the method described in [12]. These compelling results support its adoption across various applications.

REFERENCES

[1] Shawky NE. Accuracy Enhancement of GPS for Tracking Multiple Drones Based on MCMC Particle Filter. International Journal of Security and Privacy in Pervasive Computing (IJSPPC). 2020 Jan 1;12(1):1-6.

[2] A. Hechri, A. Mtibaa, Lane and Road Signs Recognition for Driver Assistance System, International Journal of Computer Science Issues, 8(6)1 (2016) 402-408.

[3] P. Harrington, Machine Learning in Action, Manning Publications Co, ISBN 9781617290183, 2017.

[4] J.G. Park, K.J. Kim, Design of a Visual Perception model with Edge-Adaptive Gabor Filter and Support Vector Machine for Traffic Sign Detection, Expert Systems with Applications, Elsevier Ltd, 40 (2013) 3679-3687.

[5] T. Bui-minh, O. Ghita, P.F. Whelan, T. Hoang, A Robust Algorithm for Detection and Classification of Traffic Signs in Video Data, International Conference on Control, Automation and Information Sciences (ICCAIS), IEEE, (2017) 108-113.

[6] R. Azad, B. Azad, I.T. Kazerooni, Optimized Method for Iranian Road Signs Detection and Recognition System, International Journal of Research in Computer Science, 4(1) (2014) 19-26.

[7] C. Zi-xing, G. Ming-qin, Traffic Sign Recognition Algorithm Based on Shape Signature and Dual-Tree Complex Wavelet Transform, Journal of Central South University Press, Springer-Verlag, Berlin, Heidelberg, 20 (2013) 433-439.

[8] F. Zaklouta, B. Stanciulescu, Real-Time Traffic Sign Recognition in Three Stage, Robotics and Autonomous Systems, Elsevier B.V., 62 (2014) 16-24.

[9] S. Wang, P. Zhang, Z. Dai, Y. Wang, R. Tao, S. Sun, Research and Practice of Traffic Lights and Traffic Signs Recognition System Based on Multicore of FPGA, Communications and Networks, SciRes, 5 (2018) 61-64.

[10] Z.L. Sun, H. Wang, W.S. Lau, G. Seet, D. Wang, Application of BW-ELM Model on Traffic Sign Recognition, Neurocomputing, Elsevier B.V., 128 (2018) 153-159.

[11] E. Oruklu, D. Pesty, J. Neveux, J.E. Guebey, Real-Time Traffic Sign Detection and Recognition for In-Car Driver Assistance Systems, IEEE 55th International Midwest Symposium on Circuits and Systems (MWSCAS), (2017) 976-979.

[12] Anurag Ranjan, Javier Romero, Michael J. Black, "Learning Human Optical Flow", Computer Vision and Pattern Recognition, (2018).