ISSN 2303-4521
Periodicals of Engineering and Natural Sciences *Original Research*
Vol. 11, No. 2, March 2023, pp.92-101

Crowd counting using Yolov5 and KCF

Mohammed Abdul Jaleel Maktoof [1], Ibraheem Nadher Ibraheem[2], Israa Tahseen Ali Al_attar[1]
[1] Computer Science Department, University of Technology, Baghdad, Iraq
[2]Faculty of Basic Education, Mustansiriyah University, Baghdad, Iraq

ABSTRACT

Crowd detection has various applications nowadays. However, detecting humans in crowded circumstances is difficult because the features of different objects conflict, making cross-state detection impossible. Detectors in the overlapping zone may therefore overreact. In this paper, real-time people counting is proposed using a proposed model of the YOLOv5 (You Only Look Once) algorithm and KCF (kernel correlation filter) algorithm. The YOLOv5 algorithm was used because it is considered one of the most accurate algorithms for detecting people in real time. Despite the high accuracy of the YOLOv5 algorithm in detecting the people in the image, video or real-time camera capturing, it needs an increase in speed.

For this reason, the YOLOv5 algorithm was combined with the KCF tracking algorithm. Where the YOLOv5 algorithm identifies people to be tracked by the KCF. The YOLOv5 algorithm was trained on a database of people, and the system's accuracy reached 98%. The speed of the proposed system was increased after adding the KCF.

Keywords: Crowd Counting, YOLOv5, Objects Tracking, KCF

Corresponding Author:

Mohammed Abdul Jaleel Maktoof
Computer Science Department, University of Technology
Baghdad, Iraq
E-mail: abdeljaleelmohammed@gmail.com

1. Introduction

Many applications of surveillance cameras, robot movement, and automatic car driving are among the most important disciplines of computer vision, which relies heavily on detecting the human body [1]. Detecting the human body is complicated, especially in crowded places, despite the significant developments in this field. In crowded places, obstruction is considered normal, which leads to difficulty in recognizing people and makes a person's visual patterns less selective [2, 3]. Two main factors that lead to the poor performance of people detection are that the object is partially hidden from view or that the visual cues are weak for the hidden object. These two factors lead to the difficulty of distinguishing objects from backgrounds or other human bodies. In addition, the visual features may differ between the overlapping objects, which is a big problem that one human being can detect in the middle of both [4-6]. There are two critical elements in smart building or smart security, which are the detection of people in addition to counting them. Each depends on using critical technologies that rely on image classification and object detection processes to discover and count people. However, many may need clarification on discovering people, such as when using videos that contain complex backgrounds [7]. Recently, deep learning algorithms have been employed, which are used in image classification and object identification, and deep learning has achieved high accuracy in these subjects [8, 9]. In 2022 [10], crowd enumeration is a valuable tool in today's culture and has applications in various fields. Among its applications is population census or human identity detection. It is a unique image processing method, and a bright future is expected.

The number of crowds has increased dramatically in recent years, so it was necessary to use deep learning techniques and artificial intelligence to detect crowds. The aim of this research was to develop a system that detects human beings in addition to dealing with counts using the YOLOv5 algorithm based on Pytorch. Where the system identifies people and sorts them for tracking and counting. In addition, the system recognizes only the required objects, which increases the system's speed, unlike other systems that rely on detecting all elements simultaneously. In 2022 [11], the research presents an application based on the Yolo and Alfi algorithm for tracking, where the system counts human beings in addition to counting bicycles. The algorithm reduces the number of frames by selecting only the necessary frames, which increases the algorithm's power and reduces memory consumption. During practical experiments, it was found that the proposed system can count and identify objects well, as the error rate reached 18 out of 525 objects. And a mean inference time of 112.82 ms per frame. With the selective down-sampling algorithm, it could also recover and reduce memory usage while maintaining its precision.

In 2021 [12], crowd-counting will be necessary for various scenarios traditionally conducted using approximate (manual) estimates and measurements. If we use deep learning, we can fix this problem. Recent crowd-counting methods typically utilize deep convolutional neural networks (CNNs) with millions of parameters to create pixel-wise density maps. These models necessitate high-performance GPUs for training, inference and usage. Since smart devices like surveillance cameras, mobiles, and Internet of things devices have limited processing capabilities, it is challenging to distribute these models to them. This work provides a novel approach to this problem with three essential components: feature fusion, Bayesian Loss, and datasets with bounding-box annotations to improve the efficiency of the crowd-counting task. To improve the effectiveness of the crowd-counting task, this study suggests a new approach based on three essential aspects: feature fusion, Bayesian Loss, and datasets using bounding-box annotations. According to the results of the experiments, the suggested method may not only allow real-time edge devices with limited processing capability but also deliver accuracy comparable to the most recent deep learning algorithms.

In 2020 [13], in this digital age, many places still use old-fashioned ways to count crowds, such as keeping registers, using people counters, and using sensors at the entrance. The areas with entirely random, highly variable, and dynamic human movements are unsuitable for these techniques. In addition to being laborious, these procedures take a lot of time. The proposed method was created for times when rapid evacuation is necessary, such as during fires, natural disasters, and other similar scenarios, as well as making intelligent decisions based on the number of people, such as food, water, congestion detection, etc. A system based on a deep convolutional neural network (DCNN) can be utilized for near-real-time crowd counting. The system utilizes the NVIDIA GPU processor and the parallel computing framework to provide rapid and agile processing of a camera's video feed. This study helps build a CCTV head detection model. The model is trained using overlapping heads, partial head visibility, etc.

This technique accurately estimates headcount in dense populations in less time. This technique accurately estimates headcount in dense populations in less time.

In 2020 [14], Major events have occurred in our world recently that has brought more attention to the significance of automatic crowd scene analysis. When a large number of people congregate in one place, as in the case of the COVID-19 breakout or a public event, it is necessary to have an automated system in place to monitor the area's inhabitants and ensure their safety. Heavy occlusion, complex behaviours, and changes in posture make analyzing crowd scenes extremely difficult. This research examines approaches for understanding congested scenes based on deep learning. The studied methods are divided into two categories: (1) crowd counting and (2) crowd action recognition. Furthermore, databases of crowd scenes are investigated. In addition to the surveys mentioned previously, this research presents an evaluation score for crowd scene analysis techniques. This measure estimates the discrepancy between the calculated and actual crowd counts in crowd scene videos.

2. Method

The proposed system is combined two algorithms to increase the accuracy of crowd counting and crowd detection. Figure 1 illustrates the central block diagram of the proposed work. Results revealed solutions to multiple issues, making the proposed development an excellent option for pinpointing population hotspots. For instance, the suggested contrast correction equation helped fix the lighting issue. Moreover, at 100 epochs, the suggested model of the YOLOv5 algorithm has achieved remarkable accuracy.

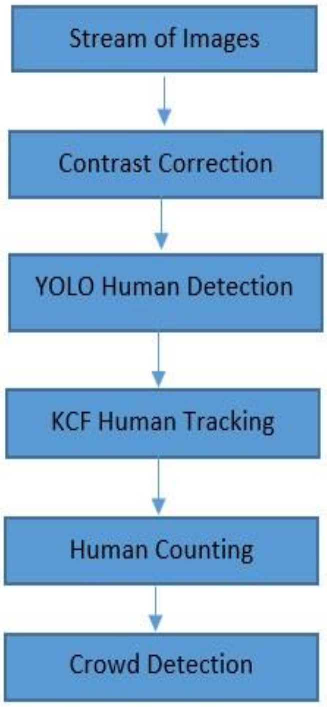

Figure 1. Block diagram of proposed work

2.1. Contrast correction

The camera capture stream of images. The proposed work computes the contrast of the image. If the contrast of an image is low, the system will increase contrast. If the contrast of capturing images is high, the system will decrease contrast. Otherwise, the image is not changed using the following proposed equations:

$$\mathrm{NewContrast(i,j)} = \begin{cases} \mathrm{Pixel(i,j)} + 20 & \mathrm{ImageContrast} < 45 \\ \mathrm{Pixel(i,j)} - 20 & \mathrm{ImageContrast} > 128 \\ \mathrm{Pixel(i,j)} & \mathrm{otherwise} \end{cases} \quad \ldots\ldots\ldots(1)$$

$\mathrm{NewContrast(i,j)}$ is the new contrast after correcting the image, $\mathrm{Pixel(i,j)}$ is the original pixel value, and $\mathrm{ImageContrast}$ is the image's contrast.

In this step, the contrast is calculated for each video frame before it is entered into the algorithm for people detection and crowding determination. The purpose of contrast correction is to improve the image to clarify the dark places and reduce the lighting of the whiter places.

2.2. Proposed YOLOv5 model

To train the YOLOv5 algorithm to detect people, a dataset was used that download from (https://www.kaggle.com/datasets/constantinwerner/human-detection-dataset). This dataset consists of two classes which are (no humans and humans). The dataset consists of 921 images (indoor and outdoor). Dividing the dataset into two parts, the training and validation parts, are included. At first, the dataset was prepared by annotating human objects from images of (human class) and then the training algorithm. The used YOLO v5 architecture consists of 24 layers and a detection layer. Figure 2 illustrates YOLO v5 architecture.

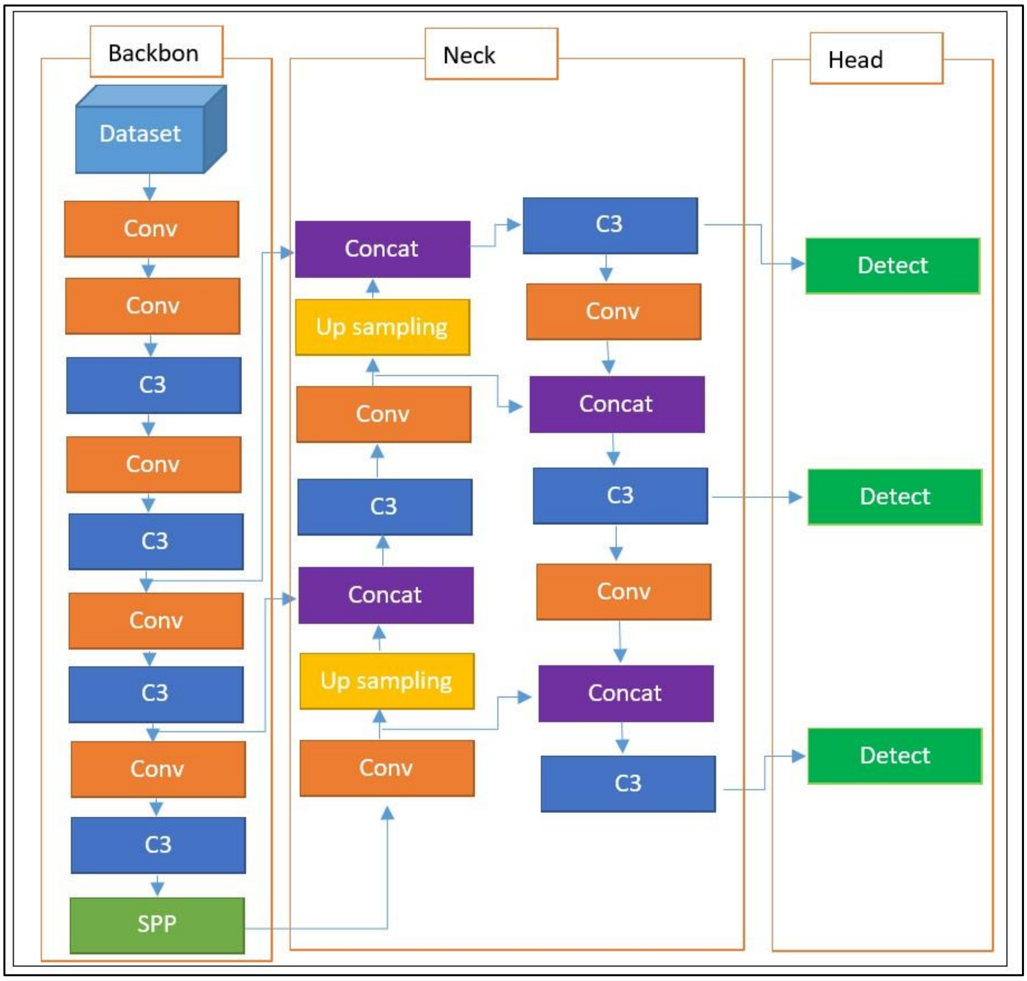

Figure 2. YOLOv5 Architecture

A new model was proposed for the YOLOv5 algorithm, where the filter sizes were changed, and the input and output sizes were manipulated to increase the prediction accuracy in detecting people. Table 1 illustrates the characteristics of the proposed model.

Table 1. Proposed YOLO v5 model layers and specifications

Part	Layer	Specifications
Backbone	Focus	[3, 60, 6, 2, 2]
	Conv	[60, 120, 3, 2]
	C3	[120, 120, 4]
	Conv	[120, 240, 3, 2]
	C3	[240, 240, 8]
	Conv	[240, 480, 3, 2]
	C3	[480, 480, 12]
	Conv	[480, 960, 3, 2]
	C3	[960, 960, 4]
	SPP	[960, 960, 5]
Neck	Conv	[960, 480, 1, 1]
	Upsampling	[None, 2, 'nearest']
	Concat	[1]
	C3	[960, 480, 4]
	Conv	[480, 240, 1, 1]

Part	Layer	Specifications
	Upsampling	[None, 2, 'nearest']
	Concat	[1]
	C3	[480, 240, 4]
	Conv	[240, 240, 3, 2]
	Concat	[1]
	C3	[480, 480, 4]
	Conv	[480, 480, 3, 2]
	Concat	[1]
	C3	[960, 960, 4]
Head	Detect	[1]

2.3. A proposed crowd-detection system

A hybrid algorithm was proposed after adding the KCK tracking algorithm to the proposed YOLOv5 model, where the proposed YOLOv5 algorithm detects people. Then the KCF algorithm tracks the people who have been detected, after which 15 frames of the video are left. The YOLOv5 algorithm will work again without taking the tracked people's locations. This addition aims to increase the proposed system's speed and increase the accuracy of detecting and counting people. Algorithm (1) illustrates the steps of the proposed hybrid algorithm.

```
Algorithm (1): A proposed hybrid algorithm
Input: Video
Output: Crowd counting
Process:
        Step 1: Load the video file.
        Step 2: Frames captures.
        Step 3: Contrast correction.
        Step 4: Apply the proposed YOLOv5 model to detect a human from a video frame.
        Step 5: Store the detected location in an array.
        Step 6: Tracking these locations using the KCF algorithm.
        Step7: Update the tracking array using the following formula:
            $\text{Tracking Array} = \begin{cases} \text{Remove location if location out of video size} \\ \text{continue} \qquad \text{otherwise} \end{cases}$
        Step8: Count the number of detection and apply the following formula:
            $\text{Result} = \begin{cases} \text{Crowd} & \text{if count} > \text{threshold} \\ \text{Not Crowd} & \text{otherwise} \end{cases}$
        Step 9: Leave 15 frames of video, then go to step 4.
        Step9: Continue until the end of the video
End
```

3. Results and discussion

The pre-processing has increased the quality of the video clip, where a proposed method was used to correct the contrast of the image, and the reason is that surveillance cameras may be placed in different atmospheres and places, so it was necessary to have an equation or algorithm that works to correct the contrast of the images. The results of the proposed equation are illustrated in Figure 3.

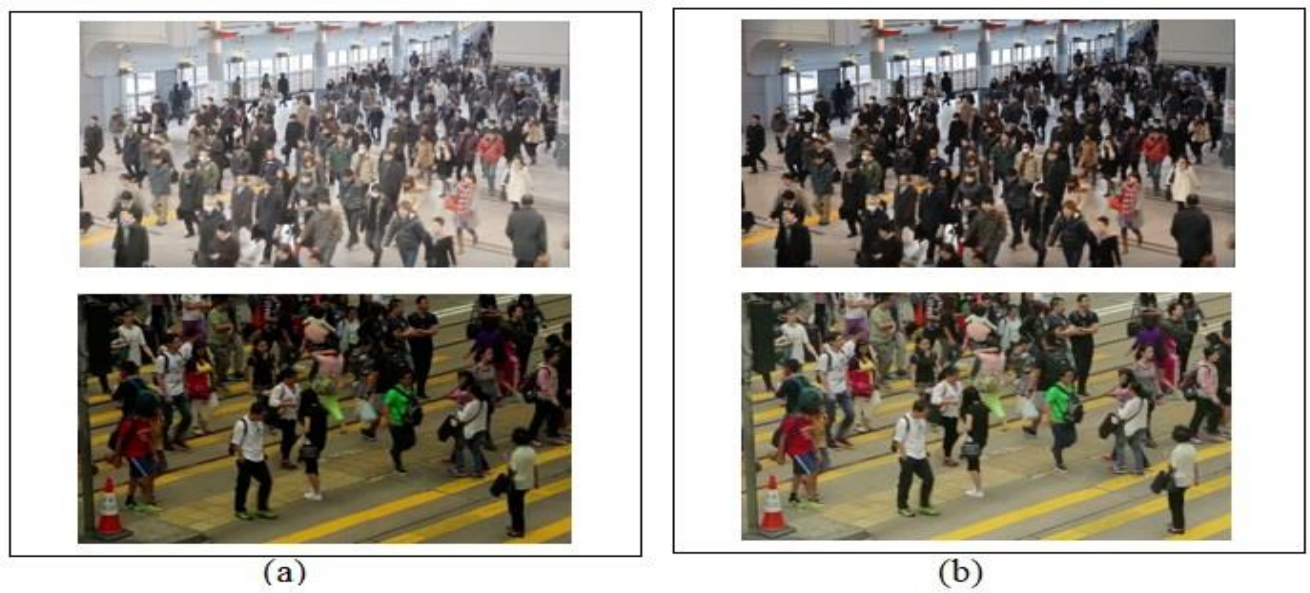

Figure 3. Contrast correction results (a) Original image (b) Pre-processed image

Many practical experiments were conducted to examine the accuracy of the proposed system. Where the proposed model was tested by training it on the (human detection) dataset, four cases were used to train the proposed work.

Case 1: when using the number of epochs 10 to train and test the proposed YOLOv5 model. Figure 4 illustrates the results of the training-testing proposed YOLOv5 model with epoch 10.

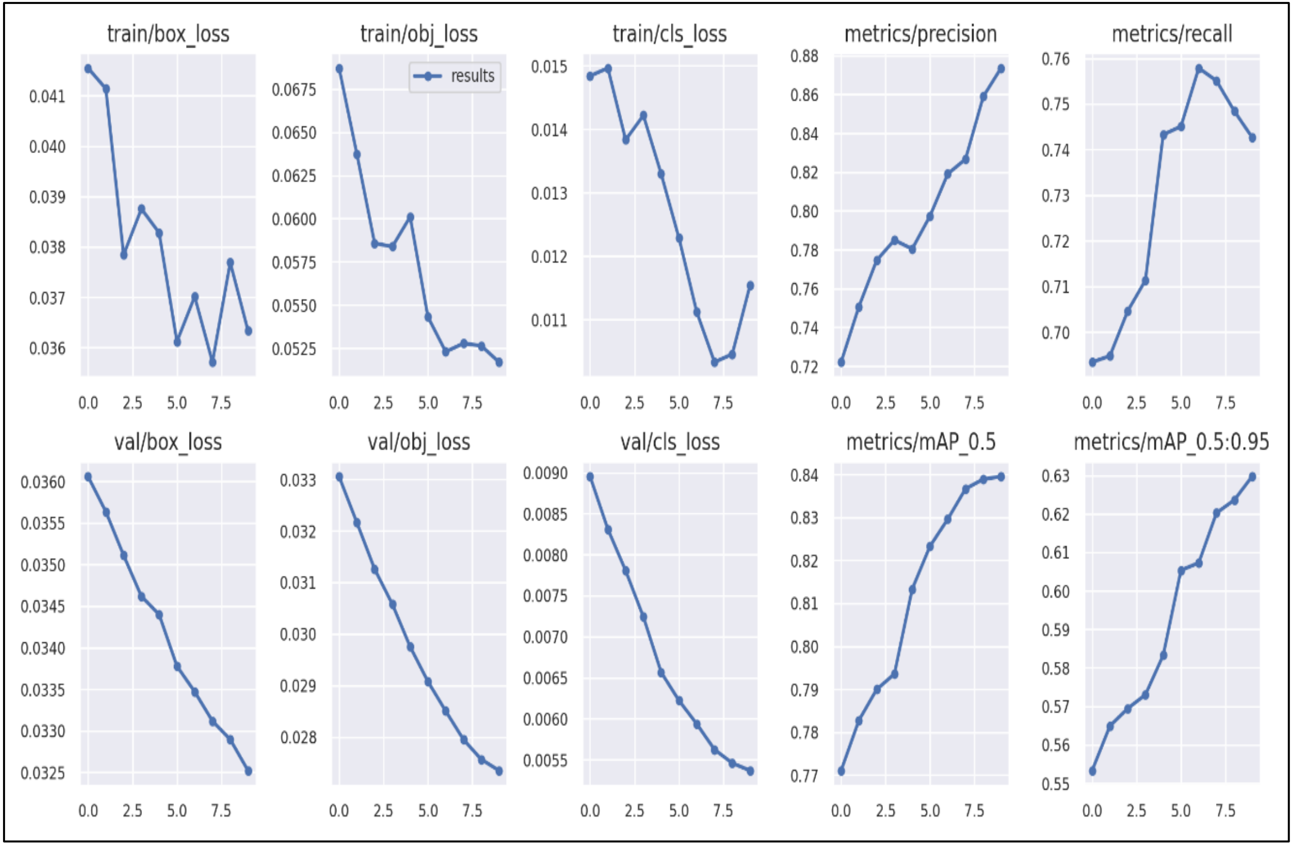

Figure 4. The results of training testing proposed the YOLOv5 model with epoch 10

Case 2: when the number of epochs is 50, train and test the proposed YOLOv5 model. Figure 5 illustrates the results of training-testing the proposed YOLOv5 model with epoch 50.

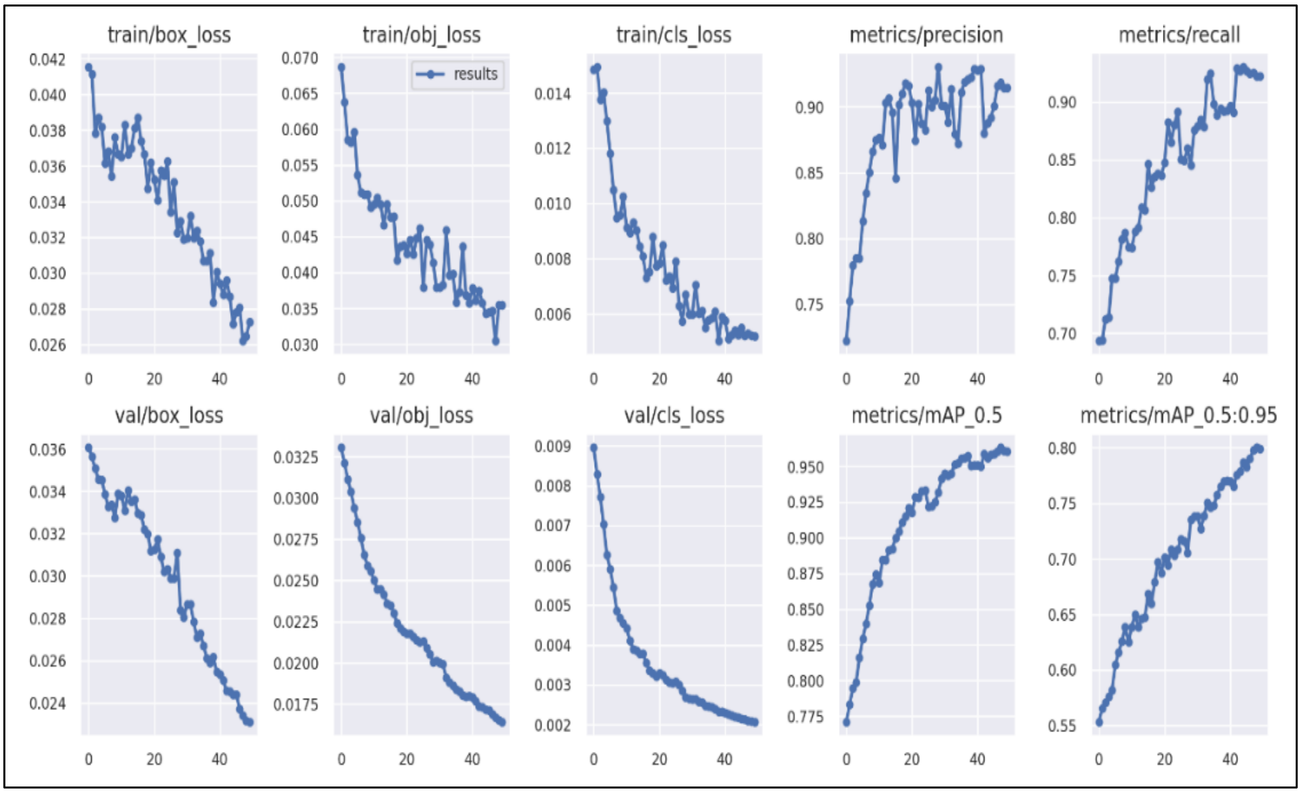

Figure 5. Results of training-testing proposed YOLOv5 model with epoch 50

Case 3: when the number of epochs is 100, train and test the proposed YOLOv5 model. Figure 6 illustrates the results of the training-testing proposed YOLOv5 model with epoch 100.

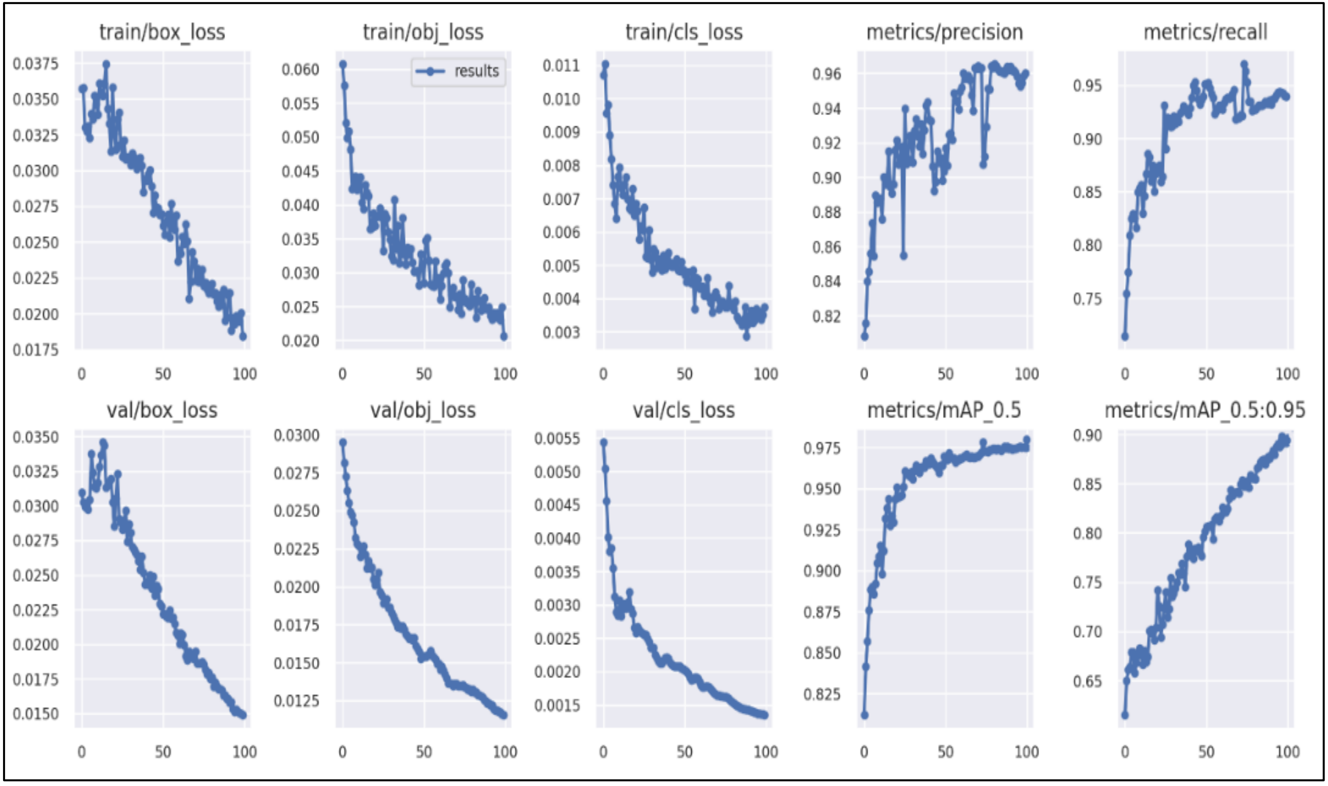

Figure 6. Results of training-testing proposed YOLOv5 model with epoch 100

Case 4: when the number of epochs is 150 to train and test the proposed YOLOv5 model. Figure 7 illustrates the results of the training-testing proposed YOLOv5 model with epoch 150.

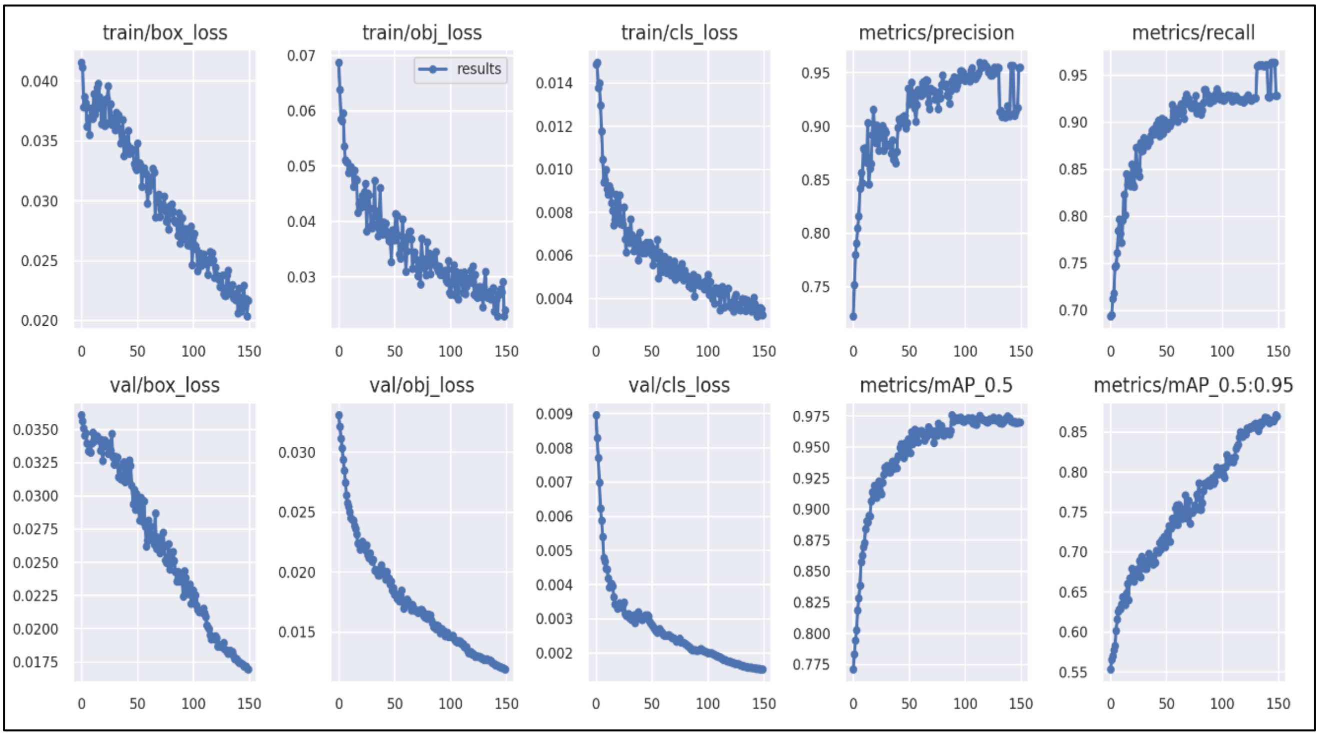

Figure 7. Results of training-testing proposed YOLOv5 model with epoch 150

The previous results can be summarized in Table 2, where the system's accuracy in training at the number of epochs 10 reached 84.21%, while the system's accuracy reached 96.37% at the number of epochs 50. Also, in the third case, when using the number of epochs 100 for training, the system's accuracy reached 97.61%. Precision is the highest resolution the system has achieved. When using the number of cycles of 150 for training, the accuracy reached 97.48%, meaning that increasing the number of epochs to a more significant percentage will lead to stability or a decrease in accuracy.

Table 2. Summary of models results

	Epoch 10	**Epoch 50**	**Epoch 100**	**Epoch 150**
Train box loss	0.0362	0.0276	0.0184	0.0212
Train object loss	0.0512	0.0352	0.0213	0.0204
Train CLS loss	0.0116	0.0054	0.0038	0.0041
Precision	0.8798	0.9257	0.0961	0.9489
Recall	0.7425	0.9354	0.9457	0.9424
Validation box loss	0.0325	0.0227	0.0150	0.0216
Validation object loss	0.0252	0.0148	0.0132	0.0134
Validation CLS loss	0.0051	0.0021	0.0014	0.0017
mAP_0.5	0.8421	0.9637	0.9761	0.9741
mAP_0.5: 0.95	0.6302	0.8001	0.9102	0.8511

3.1. System testing

To evaluate the system, a dataset called (Crowd-UIT) was used. This dataset contains ten videos taken in different places, and it was concluded that the system achieved accuracy in detecting and counting people up to more than 96 percent. Figure 8 illustrates samples of the proposed system of crowd detection, and table 3 illustrates the system evaluation for all dataset videos.

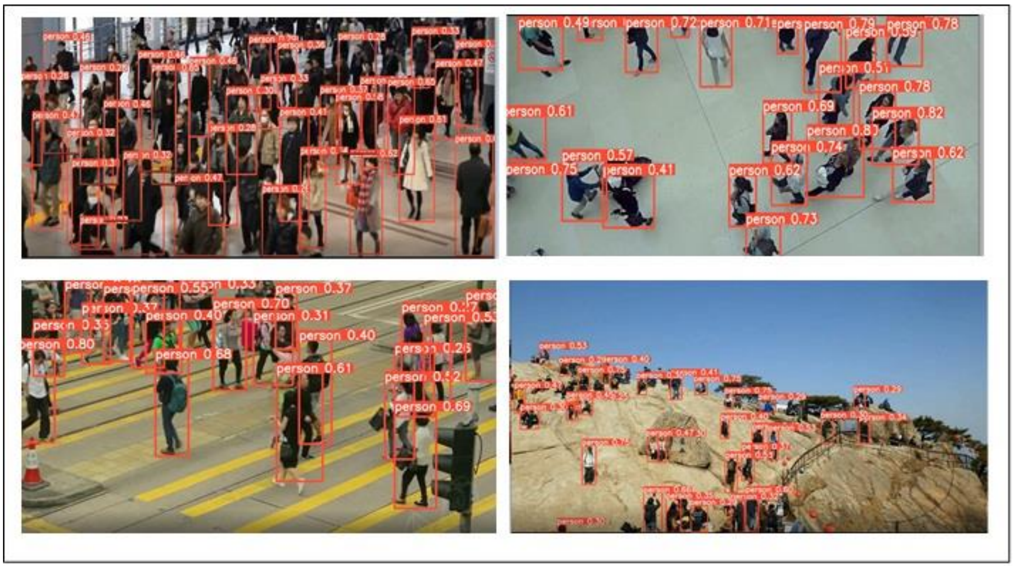

Figure 8. Crowd detection results

Table 3. Proposed System Evaluation

Video	Size	Number of People in Video	Number of Detection	Accuracy
1	1280 × 720	79	76	0.9620
2	1280 × 720	65	63	0.9692
3	1280 × 720	224	216	0.9643
4	1920 × 1080	565	551	0.9752
5	1920 × 1080	570	534	0.9368
6	1920 × 1080	207	197	0.9517
7	1920 × 1080	114	108	0.9473
8	1920 × 1080	170	163	0.9588
9	1280 × 720	72	69	0.9583
10	1280 × 720	72	68	0.9444

4. Conclusions

The problem of human crowding is one of the fundamental problems that must be considered because it may cause many accidents, such as the spread of diseases or suffocation, etc., so it was necessary to propose a system to identify crowded places to take appropriate measures. The proposed development is an excellent solution for determining the places of human congestion, as the results showed solutions to several problems. For example, the lighting problem was solved using the proposed contrast correction equation, and the results were good. In addition, the proposed model of the YOLOv5 algorithm has reached an excellent accuracy of 97.61% at 100 epochs. Also, when applying the proposed system to a Crowd-UIT dataset, the accuracy of identifying people was approximately 96% percent.

Acknowledgements
The authors thank Mustansiriyah University in Bagdad, Iraq, for cooperating with this study (http://uomustansiriyah.edu.iq).

Conflict of Interest

The authors declare that they have no conflict of interest, and they all agree to publish this paper under academic ethics.

Author Contributions

All the authors contributed equally to the manuscript.

Funding

Any official Institute or company did not support the work, which was completed by depending only on our efforts.

References

[1] H. Zhang *et al.*, "A novel infrared video surveillance system using deep learning based techniques," *Multimedia tools and applications,* vol. 77, pp. 26657-26676, 2018.

[2] W. S. Abedi, I. Nadher, and A. T. Sadiq, "Modification of deep learning technique for facial expressions and body postures recognitions," *Int. J. Adv. Sci. Technol*, vol. 29, no. 3, pp. 313-320, 2020.

[3] I. A. Aljazaery, J. S. Qateef, A. H. M. Alaidi, and R. a. M. Al_airaji, "Face Patterns Analysis and Recognition System Based on Quantum Neural Network QNN," *International Journal of Interactive Mobile Technologies,* vol. 16, no. 8, 2022.

[4] A. Antoniou and P. Angelov, "A general purpose intelligent surveillance system for mobile devices using deep learning," in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016: IEEE, pp. 2879-2886.

[5] R. ALairaji, and H. Salim, "Abnormal Behavior Detection of Students in the Examination Hall From Surveillance Videos," *Advanced Computational Paradigms and Hybrid Intelligent Computing*, vol. 1373: Springer Singapore, 2022, pp. 113-125.

[6] A. Al-zubidi, R. K. Hasoun, and S. H. Hashim, "Mobile Application to Detect Covid-19 pandemic by using Classification Techniques: Proposed System," *International Journal of Interactive Mobile Technologies,* vol. 15, no. 16, pp. 34-51, 2021.

[7] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 2147-2154.

[8] W. M. S. Abedi, A. T. Sadiq, and I. Nadher, "Modified CNN-LSTM for Pain Facial Expressions Recognition," 2020.

[9] H. T. Salim, and N. A. Jasim, "Design and Implementation of Smart City Applications Based on the Internet of Things," *International Journal of Interactive Mobile Technologies (iJIM),* vol. 15, no. 13, pp. 4-15, 2021.

[10] M. Malik, Manu Sharma, and Navya Chopra, "Crowd Counting and Detection," *International Journal of Recent Advances in Multidisciplinary Topics,* vol. 3, no. 5, pp. 49-51.2022 ,.

[11] H. Gomes, N. Redinha, N. Lavado, and M. Mendes, "Counting People and Bicycles in Real Time Using YOLO on Jetson Nano," *Energies,* vol. 15, no. 23, p. 8816, 2022.

[12] Z. Huang, R. Sinnott, and Q. Ke, "Crowd Counting Using Deep Learning in Edge Devices," in *2021 IEEE/ACM 8th International Conference on Big Data Computing, Applications and Technologies (BDCAT'21)*, 2021, pp. 28-37.

[13] U. Bhangale, S. Patil, V. Vishwanath, P. Thakker, A. Bansode, and D. Navandhar, "Near real-time crowd counting using deep learning approach," *Procedia Computer Science,* vol. 171, pp. 770-779, 2020.

[14] S. Elbishlawi, M. H. Abdelpakey, A. Eltantawy, M. S. Shehata, and M. M. Mohamed, "Deep learning-based crowd scene analysis survey," *Journal of Imaging,* vol. 6, no. 9, p. 95, 2020.