

Human Tracking System Using Optimization with Convolution Networks

¹Srilatha M and ²Srinivasu N

^{1,2}Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation,
Guntur, Vaddeswaram, Andhra Pradesh, India.

¹srilatha.manam@gmail.com, ²srinivasu28@kluniversity.in

Correspondence should be addressed to Srilatha M : srilatha.manam@gmail.com

Article Info

Journal of Machine and Computing (<https://anapub.co.ke/journals/jmc/jmc.html>)

Doi : <https://doi.org/10.53759/7669/jmc202505208>

Received 25 April 2025; Revised from 28 July 2025; Accepted 15 September 2025.

Available online 05 October 2025.

©2025 The Authors. Published by AnaPub Publications.

This is an open access article under the CC BY-NC-ND license. (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Abstract – With real-world applications in robotic vision, scene analysis, surveillance, image compression, medical imaging, and augmented reality, real-time Human Tracking Systems (HTSs) have emerged as a significant area of study in computer vision and image processing. To lessen the computational load and increase the accuracy of HTS, we present in this study a Faster Region-based Convolutional Neural Network coupled with Crow Search Optimization (FR-CNN-CSO). Additionally, to further enhance performance, an optimization strategy was combined to improve trajectory assessment, reduce individuality switches, and update functions according to environmental conditions. Moreover, the hybridized framework has provided an effective solution for improving tracking accuracy and compactional efficiency compared to conventional models. The suggested system is created using video input in a Python environment. Redundant data is eliminated from the gathered datasets during preprocessing. Histograms of Oriented Gradients (HOG) are then used for feature extraction. The extracted features are then fed into the FR-CNN model, which uses crow search optimization to effectively identify and track individuals. This strategy aims to reduce execution time and computational complexity while achieving excellent forecast accuracy. According to experimental data, the suggested approach performs better than traditional methods in terms of execution time, accuracy, precision, recall, and F-measure. The proposed FR-CNN-CSO algorithm illustrates a possible application in autonomous systems, smart surveillance systems, and provides a balance between adaptability and robustness.

Keywords – Artificial Intelligence, Crow Search Optimization, Human Tracking System, Surveillance System, Faster Region-Based Convolutional Neural Network.

I. INTRODUCTION

A comprehensive framework for identifying and tracking a reduced human model is known as "human tracking," and it usually makes use of synchronized monocular grayscale image sequences taken from one or more camera positions [1]. The approach mostly entails locating the target individual in a video scene and tracking their movements over time. Vision-based tracking and detection systems have gained popularity recently as a way to improve security, convenience, and safety [2]. For this reason, human tracking and detection are seen as crucial elements of contemporary surveillance systems. Moving object extraction and human recognition are often the two primary responsibilities of a human detection system [3,4]. Separating foreground items from the background and determining their size and spatial placement within the film are the main goals of moving object extraction [5]. A picture is classified as either human or inhuman by human recognition. The device that tracks must be able to predict the position before and after occlusion, even when the monitored individual or object may be concealed by other objects while being followed [6]. Active and fixed cameras are the two types commonly seen in surveillance systems. Because it can tilt and pan to keep the target object in the frame, an active camera has a greater field of view (FOV) than a fixed camera, which has the added benefit of being less expensive but having a smaller FOV [7, 8]. Additionally, the latter's zoom-in/zoom-out capabilities allow it to provide a higher resolution.

To extract moving objects, a monitoring system using an active camera frequently takes the temporal difference into account. For processing the image in this manner, the camera needs to be sufficiently stable [9, 10]. To put it another way, the camera isolates background pixels from the images it takes as an object moves [11]. The active camera behaves erratically and jerkily as a result. This problem is addressed by an optimization technique [12]. Initially, the method of deep learning is used to identify the human as the target model. The optimization method then calculates the distance between the color histogram of the target model and the succeeding color histogram form of the chosen position to track

the individual [13, 14]. Among the many advantages of a color histogram are its rotation, scale invariance, non-rigid object tracking, partial occlusion resistance, and speed of computation [15].

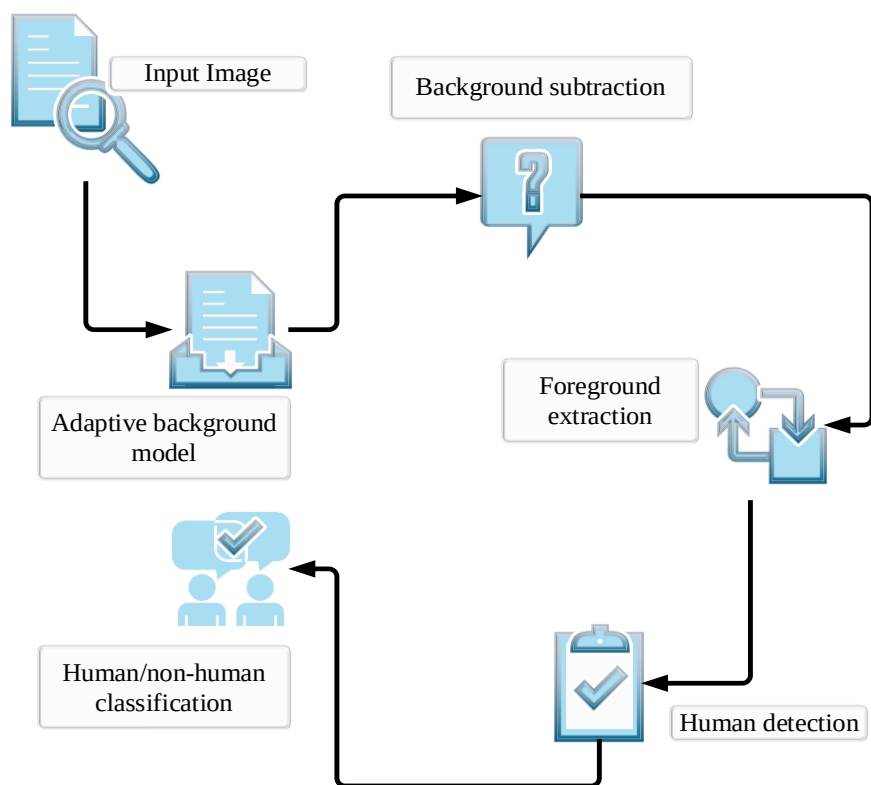


Fig 1. Tracking System of The Human.

Fig 1 depicts the human tracking system's surveillance mechanism. The fast growth of intelligence-based applications has led to the widespread usage of intelligent systems for information processing in modern civilization [16]. One of the most popular research topics these days is the Internet of Things and Artificial Intelligence, which integrates IoT architecture and AI technologies to deliver more effective IoT services [17]. It has proven to be highly effective for a range of monitoring applications in complex industrial settings, both indoors and outdoors [18]. However, abruptly congested scenes, motion changes, camera angles, close interpersonal contact, and occlusion may negatively impact the effectiveness of human detecting techniques [19]. In addition, there are significant differences in the individual's physical attributes, including height, posture, articulation, scales, rotation, movements, sizes, and torso. Robot vision, surveillance, tennis ball detection in sports, traffic monitoring, animation, sports performance analysis, and motion tracking are only a few of the numerous applications [20, 21]. Several methods for tracking people have already been presented. However, it is shown to be a crucial task that does not yield results that are 100% precise in identifying and continuously tracking humans. Furthermore, human safety is a critical problem given the sharp increase in accidents and crimes [22, 23].

Detecting persons due to posture and perspective fluctuation, computational complexity, occlusion, classification, and regression problems is the most difficult challenge in human tracking systems [24]. To improve the human tracking system's performance, a number of deep learning models were created, including Region-based CNN (R-CNN), YOLOv2, and Convolutional Neural Networks (CNN). However, because to the complexity of the data, the error rate, and the limited number of prediction results, a satisfactory solution was still not identified [25]. To improve the human tracking system, create a Faster Region-based Convolutional Neural Network (FR-CNN) with Crow Search Optimization (FR-CNN-CSO) architecture that can precisely identify people from the surveillance data gathered. Additionally, the CSO fitness model, which continuously scans the monitoring system and forecasts human behavior, is used to precisely monitor and track humans. The proposed model's primary goal is to enhance the human tracking system through optimization techniques and get improved prediction outcomes.

Main Objectives

- To use deep learning and CSO to create an efficient person tracking system.
- To use a faster region-based CNN to enhance the prediction outcomes of a collection of videos.
- To demonstrate the effectiveness of the developed model by achieving improved classification and feature extraction outcomes.

Furthermore, the suggested technique's crucial contribution is explained in full below.

- To start, the system was trained using a video dataset that was gathered from the internet.
- In addition, create a FR-CNN-CSO framework that can precisely identify and categorize a human tracking system.
- After that, input videos undergo preprocessing to eliminate unnecessary data from the gathered datasets.
- From this point on, feature extraction is used to extract pertinent features using histograms of directed gradients.
- After that, the retrieved features are updated into the FR-CNN model, which uses crow search fitness to identify humans with high performance.
- Lastly, to improve the prediction results, update the CSO fitness in the classification layer. The obtained findings are then verified using other popular models.

The structure of the manuscript is as follows: problem statements and related works are described in sections 2 and 3. The suggested methodology was explained in detail in Section 4, and the findings and comparison of the suggested strategy with previous studies were covered in Section 5. Section 6 concludes and future directions.

II. RELATED WORKS

Ming-Chuan et al. [26] created a DL-based system that integrates two object detection methods: region-based convolutional neural networks (R-CNN) and quicker mask-based convolutional neural networks. It is used to find the target's motion reference point and to gather the segmentation. After combining the data from the two methods, the location of the object and its actual rotated angle are ascertained. The accuracy of the detection was 96.26%, according to the data. Processing time is also shortened by increasing production capacity.

An automatic person detection system was proposed by Imran Ahmed et al. [27] to track and identify people in a complex industrial environment. A top view is usually chosen because it can provide sufficient coverage and vision of a situation. The applicability, effectiveness, and viability of DL designs utilizing transfer learning are demonstrated by this study. The effectiveness of every detection design has significantly increased, with the greatest True Positive Rate (TPR) of 93% demonstrating encouraging outcomes.

DL-based computer vision is utilized to determine the person's condition and state. Kamel Boudjit et al. [28] created algorithms for identifying and recognizing individuals using a convolutional neural network (CNN) YOLOv2, which are dependent on a drone's cameras. The person detection results show that YOLO-v2 has a high degree of object recognition and categorization accuracy. Moreover, the tracking algorithm responds more quickly for real-time tracking than popular techniques, successfully following the identified individual without taking them out of view.

Hiten Goyal et al. [29] have published a face mask detection system that can distinguish between photographs with and without a mask in both static photos and live videos. The model is trained and evaluated using the Kaggle data set. There are about 4,000 photos in the gathered data set, with a 98% performance accuracy grade. In terms of computation, the proposed model is precise and effective.

To detect small objects, Amudhan A.N. et al. [30] developed a hypermetropic CNN. To improve detection, low-level features are moved to the deeper layers, and more features are taken out of the shallow levels. The system is hypermetropic since it performs better on nearby items and worse on farther ones. Significant improvements in small-sized object detection, including a 32% increase in false-favorable rates, are demonstrated by the proposed method.

III. PROBLEM DEFINITION

The useful human tracking system reduces processing time and increases production capacity. Enhancing computational efficiency is the most difficult job in the HTS [31]. Confined and regulated environments are generally better for AI-based system training. After been trained in a variety of scenarios with relevant data variations, it can be used with good accuracy in a different context. Provide the AI technique for learning the components of the target item rather than the environment's aspects. Data-hungry DL systems [32]. Pose variation, viewpoint fluctuation, and occlusion are only a few of the many issues that arise while detecting human tracking systems. The most frequent challenges include occlusion, error rate, computing time, vanishing gradient problems, and classification and regression problems. The goal of these problems is to increase computing efficiency in order to improve the performance of individual tracking systems.

IV. PROPOSED METHODOLOGY

To improve prediction efficiency and accuracy, the researchers have suggested combining deep learning with an alternative metaheuristic method. This method can tackle various picture problems and produce superior results. As a result, this paper's preprocessing involves gathering and updating input videos.

Fig 2 depicts the structure of the suggested model. Eliminate superfluous information from the gathered datasets during the preparation phase. Next, use HOG to extract the pertinent features in the dataset to enhance prediction outcomes. Then, for precise human tracking and recognition, create a new Faster Region-based Convolutional Neural Network with Crow Search Optimization (FR-CNN-CSO) architecture. In the fully connected layer of the FR-CNN model, it then modifies Crow Search Optimization (CSO) [33] for accurate identification and to increase computational complexity. Lastly, the simulation results show the benefit of FR-CNN-CSO in terms of classification accuracy for human tracking systems.

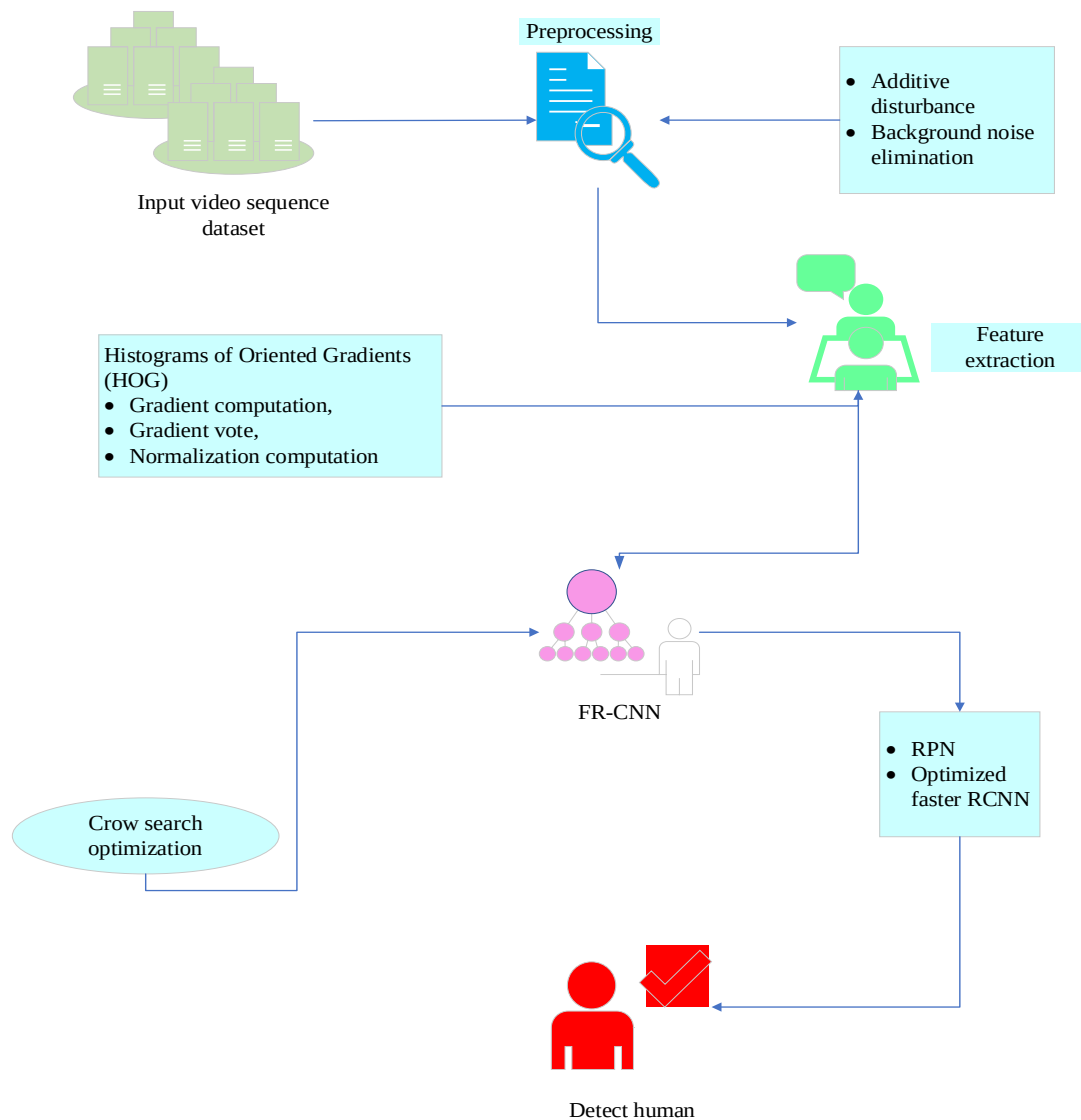


Fig 2. Proposed Methodology of FR-CNN-CSO.

Dataset Description

Challenging Events in Person Detection from Overhead Fisheye Images (CEPDOF) (<https://vip.bu.edu/projects/vsns/cossy/datasets/cepdof/download/>) is used in the study. The new dataset includes 8 videos with up to 13 people visible at once, over 25,000 evaluated frames per second, extreme body occlusions, crowded spaces, different body positions, head camouflage images of people in a projection screen, dim lighting with or without Infra-Red (IR) illumination, and several novel challenging situations.





Fig 3. Sample Images of The Used Dataset.

Furthermore, bounding boxes of the same individual in later frames will have the same ID due to the spatiotemporal marking of CEPDOF. It can therefore be used for further vision tasks, such as re-identifying individuals from fisheye overhead photographs and tracking moving objects in movies. Furthermore, 20% of the dataset is used for testing, while the remaining 80% is used for training. **Fig 3** displays the dataset's sample photos.

Preprocessing

To differentiate between noisy and undetectable datasets, preprocessing is the initial step in generating vectors of features in an HTS. The collected input datasets $I(n)$ are adjusted or modified during preprocessing to make them more suitable for feature extraction examination.

Checking for errors and disturbances in the $I(n)$ data is the main thing consider regarding human detection processing. Because $I(n)$ is corrupted by some background noise $F(n)$. Equation (1) is used to determine the additive disturbance.

$$I(n) = v(n) + F(n) \quad (1)$$

Let, cleaned dataset be represented as $v(n)$. The work is performed on noisy data using a variety of noise reduction techniques. However, the two noise reduction methods most frequently used in HTS to create HTS are adaptive cancellation of noise and spectral subtraction.

The ambient noise must be taken into account while adjusting the noise levels in HTS. The performance metric of human identification systems is significantly impacted when test and training data are collected with different noise levels. The creators of HTS usually do not influence the noises that occur in real-world scenarios. A popular method for reducing the effect of noisy surroundings on human recognition during both the training and testing stages is preprocessing. To get rid of background noise, Eqn. (2) uses the following filter.

$$B_{no} = 10 * \log_{10} \left[\theta + \frac{1}{M} \sum_{n=1}^M v^2(n) \right] \quad (2)$$

Let, B_{no} is measured as the elimination of background noise using M samples, a small positive constant is represented as θ .

Feature Extraction

Histograms of Oriented Gradients (HOG) are used to extract the pertinent features from the pre-processed dataset during the feature extraction phase [34]. It is used to extract features from video or picture data. It is frequently employed in computer vision tasks for object detection. Additionally, the three stages of HOG feature extraction include gradient computation, gradient vote, and normalization computation.

The gradient computation is extracted based on the coordinate pixel (a, b) , direction $\mu(a, b)$, and magnitude $m(a, b)$ is calculated in this phase. Equations (3) and (4) are used to measure the gradient and luminance values of a and b .

$$f_a(a, b) = f(a+1, b) - f(a-1, b) \quad (3)$$

$$f_b(a,b) = f(a,b+1) - f(a,b-1) \quad (4)$$

Let, $f_a(a,b)$ and $f_b(a,b)$ are considered as the gradient of a and b axes. Moreover, $f(a,b)$ is denoted as a luminance value.

The magnitude $m(a,b)$ is measured using Eqn. (5).

$$m(a,b) = \sqrt{f_a(a,b)^2 + f_b(a,b)^2} \quad (5)$$

The direction $\mu(a,b)$ is computed using Eqn. (6).

$$\mu(a,b) = \arctan \frac{f_b(a,b)}{f_a(a,b)} \quad (6)$$

After obtaining the magnitude $m(a,b)$ and direction $\mu(a,b)$, each pixel within the cell produces a gradient vote for a vertically oriented histogram based on the orientation of the gradient element that surrounds it. Nine bins are uniformly distributed throughout the 0–180 orientation range. Eqn (7) can be used to determine the weight of each pixel, which is represented by the symbol.

$$\tau = (j + 0.5) - \frac{c * \mu(a,b)}{\pi} \quad (7)$$

Let, j is represented as the bin to which $\mu(a,b)$ belongs and the total number of bins is denoted as c . Finally, all the histograms from a single block with four cells are collected to generate a histogram normalization computation. Following normalization, the outcome can be expressed as Eqn (8).

$$g_i^2 = \frac{g_i}{\sqrt{\|g\|_2^2 + \lambda^2}} \quad (8)$$

Let, several cells and bins is denoted as i , g_i is considered a vector corresponding by combining histogram for block region, and small constant is denoted as λ .

Human Detection and Tracking Using FR-CNN-CSO

Faster R-CNN (FR-CNN) [35] is a more conventional deep learning method that provides a higher detection performance, effectiveness, and enhanced recognition rate to a significant target region. The Faster R-CNN technique is composed of two main parts: the extraction module and the detection module. The FR-CNN model is the detection module, and the Region Proposal Network (RPN) is the extraction module. The RPN is used from the baseline feature map to generate high-quality regions, and the Fast R-CNN instantly identifies and categorizes the targets in the derived suggestions region. Photos of any size are stored on the VGG-16 network. Second, the shared convolutional layer or feature map was generated by the CNN network. The feature map is spread to a specific convolutional layer after being introduced into the RPN network, producing a higher-dimensional map of features. Finally, the RoI Pooling was used to extract the recommended region's attributes from the higher-dimensional feature map. **Fig 4** displays the architecture within the optimized FR-CNN model.

The features were then incorporated into the regression and classification layers. CSO optimization was used to enhance the HTS prediction results. The target object category & the Region's coordinates were then supplied by the algorithm. Deep learning effectiveness has greatly improved, and the faster R-CNN method has demonstrated encouraging results in target identification and recognition.

Crow Search Optimization

The Crow Search Algorithm (CSA), a novel population-based algorithm, imitates how crows conceal food. Intelligent birds, crows, can identify faces and warn their species of danger. They demonstrate their slyness, for example, by concealing food and keeping track of its location. A fitness function is used to evaluate each crow, and the result is saved as the starting memory value. Every crow changes its location by selecting a random crow and logs its hiding place in its

memory parameter. Based on their actions during random selection, this fitness function is used to find and track human participants in the dataset. This increases the computational complexity and improves the efficacy of the person tracking system.

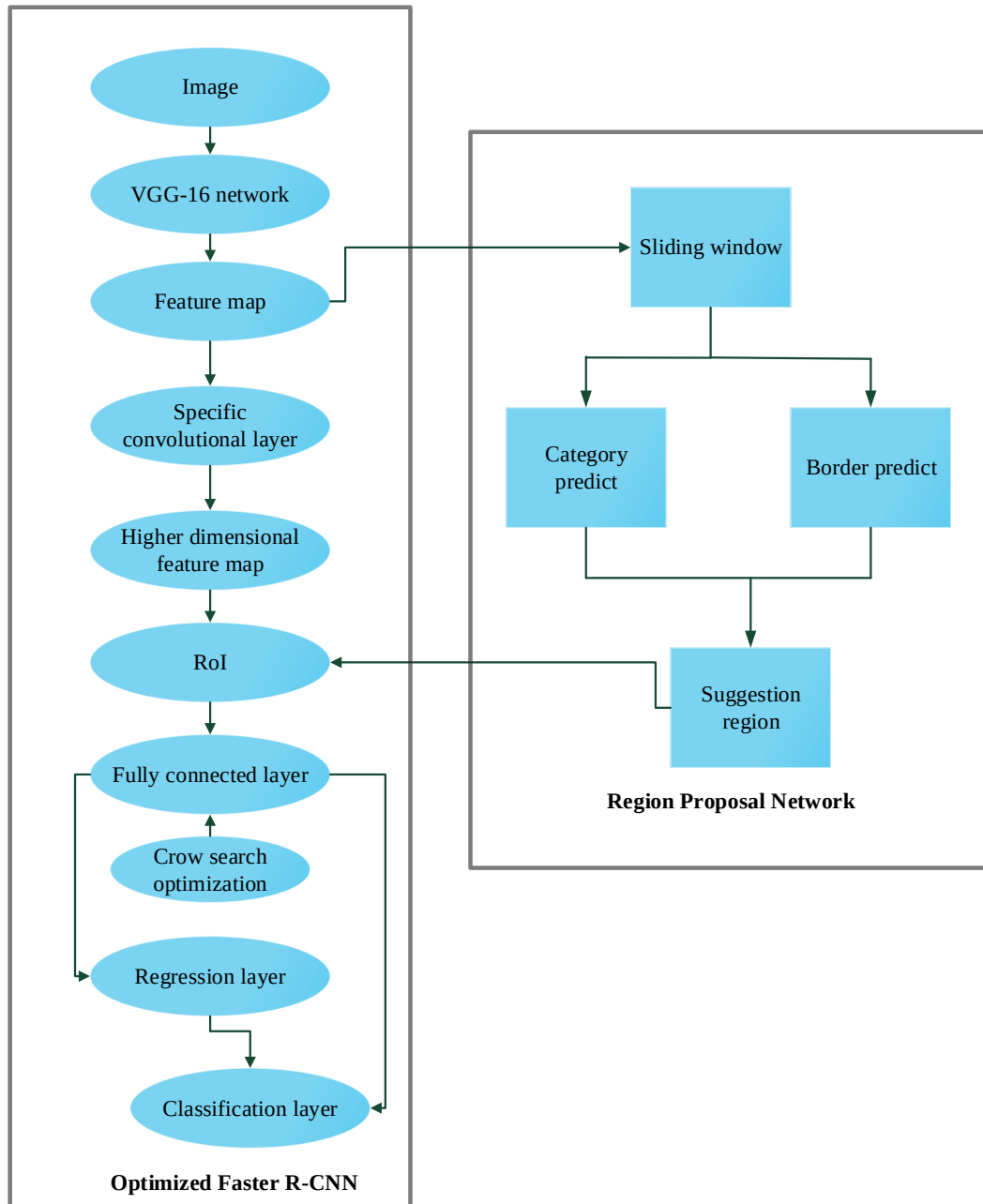


Fig 4. Architecture of Optimized FR-CNN Model.

First, use Faster R-CNN in person detection to increase computational complexity. The output contains the class title and the confidence score value for the rectangle bounding box's height, coordinates, and width. This boundary box contains the information about the object's location on the video. RPN, a two-stage detector, is used in the first step to create region anchors. Using the identified anchor regions, the next step gathers bounding box data for human classification. The input video's properties are obtained through convolution layers. The retrieved features are then used to further develop the generated feature maps. The sliding window mechanism is used to generate anchor or area boxes. The appearance of an object or person in the film is identified by further refining these anchor boxes. In the final step, the anchors/detected boundaries are enhanced. Additionally, a CSO fitness function is used in a fully connected layer to enhance human prediction using a tiny network, and the optimal anchor areas are determined by computing the loss function.

From an input video (size), the RPN produces a sequence of rectangular object proposals, each of which has an object score. Additionally, use the network to extract convolutional features from the source images. Using a $n \times n$ spatial window as input, the convolutional feature maps are fed into a small network that converts them from a lower-dimensional

set of features. This feature is put into a box and a regression layer that establishes the box as a set of classification tasks or backdrop. To improve detection accuracy, place reference boxes (anchors) at each sliding-window location with measurements and aspect ratios. Three scales & three aspect ratios resulting in $s = 9$ at each sliding position are shared by the Faster R-CNN anchors. Eqn. (9) is used to assess the loss of the RPN function (9).

$$A(\{Y_i\}, \{Q_i\}) = \frac{1}{n_{cl}} \sum_i A_{cl}(Y_i, Y_i^*) + \eta \frac{1}{n_{rg}} \sum_i Y_i^* A_{rg}(Q_i, Q_i^*) + C_s(t) \quad (9)$$

Where, anchor index is represented as i , Y_i is denoted as anchor predicted probability belonging to i object, vector representation of predicted boundary box is denoted as Q_i , and Q_i^* is called ground truth. Moreover, n_{cl} is denoted as minibatch size, several anchor locations is represented as n_{rg} , and $C_s(t)$ is represented as fitness of the CSO.

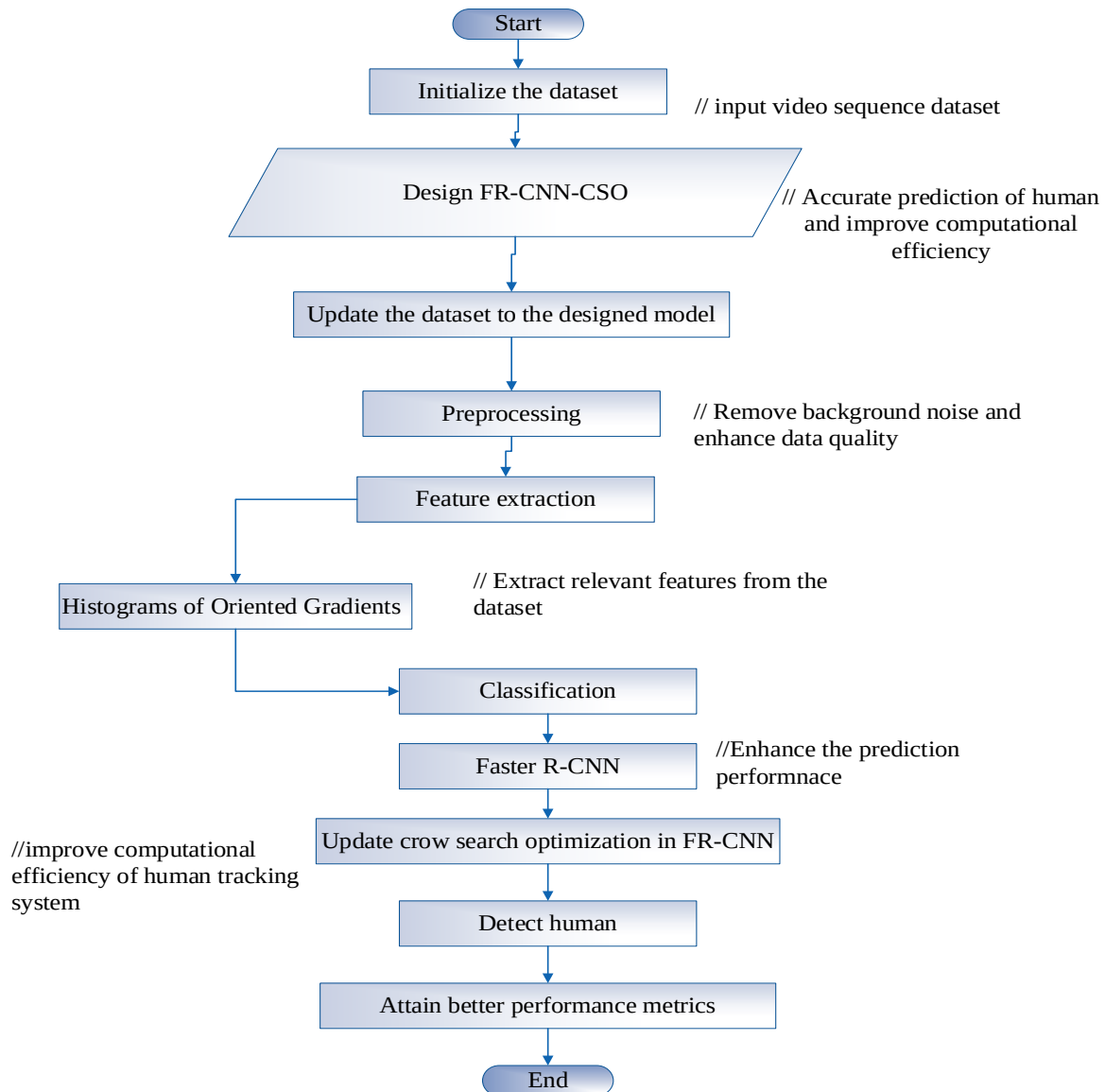


Fig 5. FR-CNN-CSO Model Flowchart.

Equation (10) describes the coordinates for the boundary box regression parameterizations.

$$\begin{aligned}
Q_x &= (X - X_b) / D_b, Q_y = (Y - Y_b) / E_b, \\
Q_d &= \log\left(\frac{D}{D_b}\right), Q_e = \log\left(\frac{E}{E_b}\right), \\
Q_x^* &= (X^* - X_b) / D_b, Q_y^* = (Y^* - Y_b) / E_b, \\
Q_d^* &= \log\left(\frac{D^*}{D_b}\right), Q_e^* = \log\left(\frac{E^*}{E_b}\right),
\end{aligned} \tag{10}$$

Where, box center coordinates are represented as X , and Y , box centre's width and height is considered as E , predicted box is represented as X , anchor box is represented as X_b , and ground truth box is represented as X^* . Finding and classifying every instance of a human within a bounding box in the data provided image or video frames is the aim of a human tracking system. The loss function is used in both classification and regression. Lastly, the output displays the bounding box with persons and the overall class score. **Fig 5** displays the created model's flow chart.

V. RESULTS AND DISCUSSIONS

Python software is used in the FR-CNN-CSO methodology to track and find individuals. To increase computational complexity and improve the precision of the findings, the Faster R-CNN modifies the CSO fitness. Additionally, the dataset's noise is removed during the preparation phase. Additionally, feature extraction takes the dataset's most important information while eliminating unnecessary features. In the end, a constructed model can recognize and classify individuals with speed and accuracy.

Performance Analysis

To show the effectiveness of the devised method, the acquired results are compared to other commonly used traditional techniques. Additionally, performance metrics such as execution time, recall, accuracy, precision, and F-score are compared. Thus, YOLOv3 [27], YOLO-v2 [28], and CNN [29] are the methods now in use.

Accuracy

Accuracy is evaluated using closed assessments to determine the acceptable or true value. Accuracy is used to determine something's actual or desired worth. The accuracy is calculated using Equation (11).

$$A_c = \frac{t_{po}^* + t_{ne}^*}{t_{po}^* + f_{po}^* + t_{ne}^* + f_{ne}^*} \tag{11}$$

Where t_{po}^* denoted as the actual positive rate of a detected human, t_{ne}^* is considered the correct negative rate of a detected human, f_{po}^* and f_{ne}^* are denoted as the incorrect favourable and incorrect negative rates of detected humans. The comparison of the accuracy is exposed in **Fig 6**.

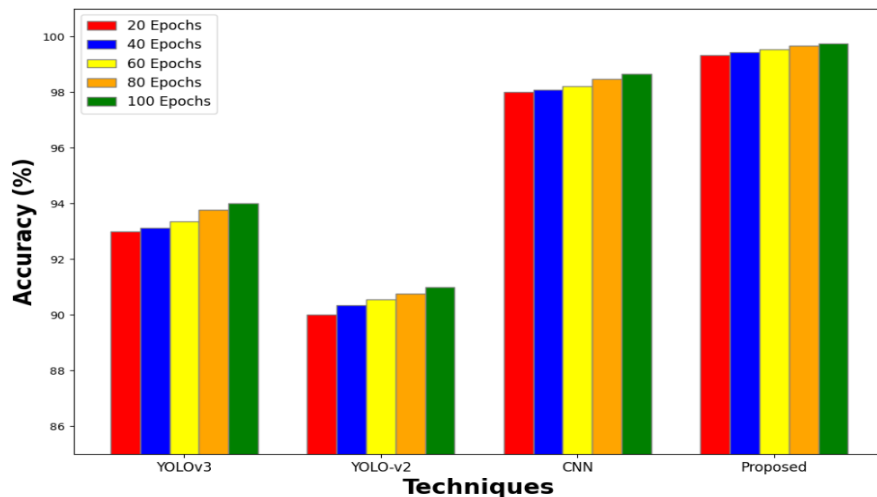


Fig 6. Comparison of Accuracy.

The suggested model's improved accuracy results are contrasted with those of well-known models such as CNN, YOLOv3, and YOLO-v2. Additionally, for 20 epochs, the accuracy rates of YOLOv3 and YOLO-v2 reached 93% and 90%, respectively. The CNN model had comparable outcomes (98% accuracy). After 20 epochs, the developed model achieved an accuracy of 99.34%. Over 100 epochs, the accuracy rates for YOLOv3 with YOLO-v2 reached 94% and 90.98%, respectively. The CNN model produced comparable outcomes (98.66% accuracy). In the end, the developed model achieved a 99.75% accuracy rate over 1000 epochs. When compared to other models, the designed technique receives greater accuracy ratings.

Precision

Furthermore, precision is assessed using the human detecting system's closed metrics. When measured by consistent outcomes, precision is more reliable than accuracy. Eqn. (12) is used to calculate precision.

$$P_r = \frac{t_{po}^*}{t_{po}^* + f_{po}^*} \quad (12)$$

Results from other well-known models, such as CNN, YOLOv3, and YOLO-v2, are used to validate the enhanced precision of the created model. Over 20 epochs, the precision rates for YOLOv3 and YOLO-v2 increased by 90% and 89.41%, respectively. Accordingly, during 20 epochs, the CNN approach achieves 98% precision. After 20 epochs, the created model finally reaches a 99.45% precision. Additionally, during 100 epochs, the precision rates for YOLOv3 and YOLO-v2 increased by 91.76% and 90.76%, respectively. Accordingly, for 100 epochs, the CNN approach achieves 98.47% precision. Finally, during 100 epochs, the created model attains a precision of 99.79%. The designed model receives higher precision scores than the others. A graphical illustration of precision is shown in Fig 7.

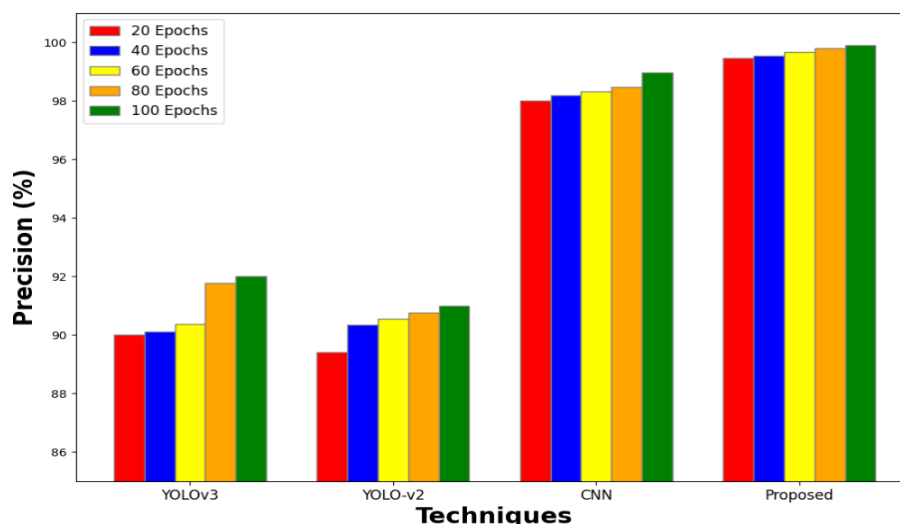


Fig 7. Comparison of Precision.

Recall

The recall is the sum of all pertinent information that has been found and is useful in some way. Recall is the capacity of the developed model to locate every suitable instance in the dataset. The calculation also involves dividing the entire amount all true positives with the sum of the true positives and false negatives. Eqn (13) is used to calculate the recall.

$$R_e = \frac{t_{po}^*}{t_{po}^* + f_{ne}^*} \quad (13)$$

Several popular models, such as CNN, YOLOv3, and YOLO-v2, are utilized to validate the constructed model's acquired recall findings. Additionally, during 20 epochs, the recall rates for YOLOv3 and YOLO-v2 reached 88% and 83.5%, respectively. The CNN model produced comparable outcomes (97%) over 20 epochs. In the end, the developed model achieves 99.15% recall for 20 epochs. Furthermore, during 100 epochs, the recall rates for YOLOv3 and YOLO-v2 increased by 89% and 86%, respectively. The CNN model produced comparable outcomes (86%) over 100 epochs. In the end, the developed model achieves 99.69% recall for 100 epochs. The designed model achieves greater recall ratings than the others. A visual representation of recall is shown in Fig 8.

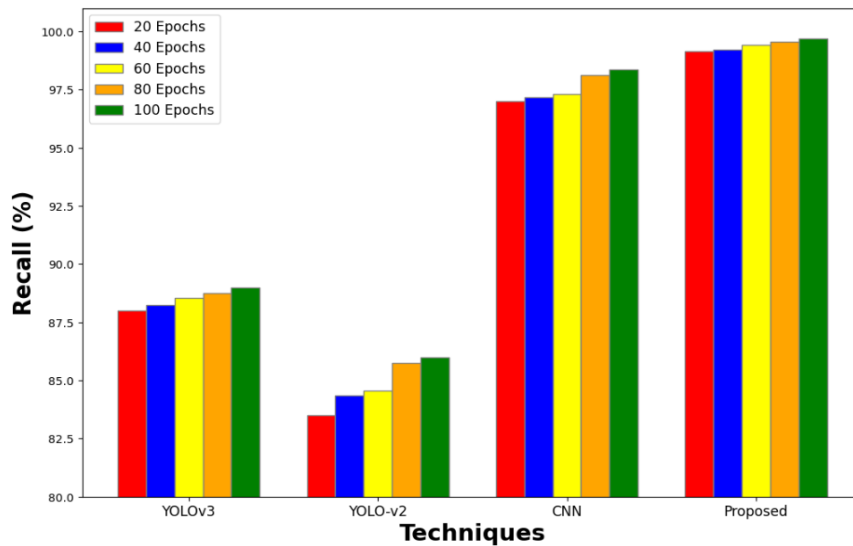


Fig 8. Comparison of Recall.

F-Score

The sum of all the classification measures yields the F-score. The F-score evaluates the scoring method's recall and accuracy. The accuracy of the classification measures as predictors is shown in the higher F-score. The F-score is calculated using Equation (14) as a guide.

$$F_s = 2 \times \frac{P_r \times R_e}{P_r + R_e} \quad (14)$$

Where, precision is represented as P_r and recall is represented as R_e .

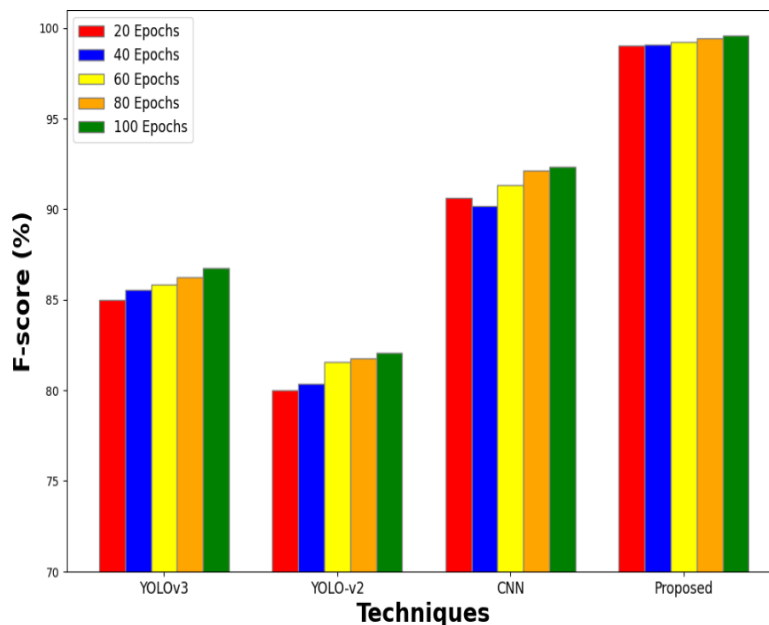


Fig 9. Comparison of F-Score.

The F-score findings of the created model are verified against CNN, YOLOv3, YOLO-v2, and other popular models. For 20 epochs, the F-score rates of YOLOv3 and YOLO-v2 increased by 85% and 80%, respectively. Accordingly, the CNN method obtains an F-score of 90.6% for 20 epochs. The designed technique eventually achieves a 99% F-score after 20 epochs. Furthermore, during 100 epochs, the F-score rates of YOLOv3 and YOLO-v2 increased by 86.76% and 82.05%, respectively. Accordingly, the CNN method obtains an F-score of 92.35% for 100 epochs. Ultimately, the developed method achieves a 99.57% F-score over 100 epochs. The designed model has greater F-scores than the others. Furthermore, when compared to other models, the created one had higher F-scores. The F-score is represented graphically in Fig 9.

Execution Time

The duration of a process is determined by its execution time, which is usually independent of its start time but often depends on the incoming data. The execution time is the interval across the start and finish times. Subtract the beginning time from the ending time to determine the execution time. The exponential pattern of execution times as the input size gets closer to infinity is known as time complexity. Computational complexity quantifies how long an algorithm takes to run, given the length of the input. It calculates how long each algorithm's program statement will take to execute. **Fig 10** displays an illustration of the execution times.

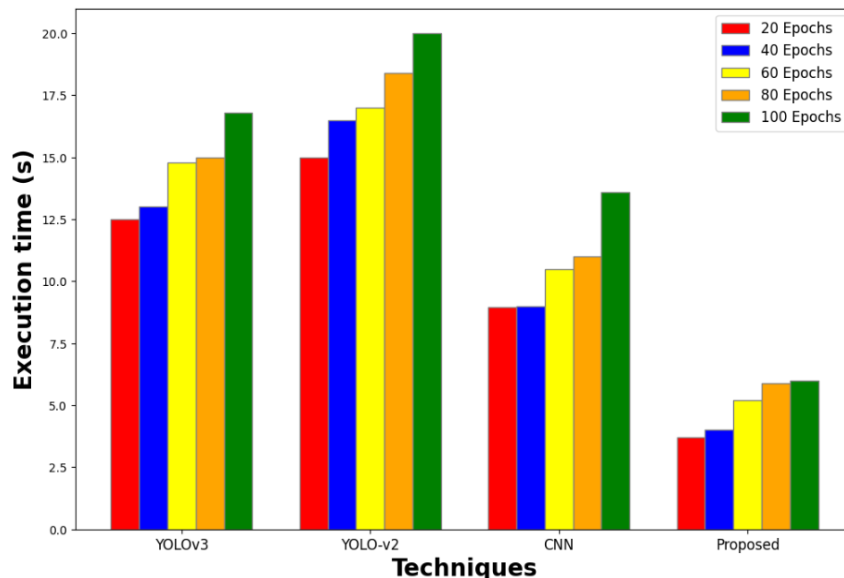


Fig 10. Comparison of Execution Time.

The acquired execution time results of the suggested model are compared to those of other popular models, such as CNN, YOLOv3, and YOLO-v2. For 20 epochs, YOLOv3 and YOLO-v2 had execution times of 12.5 and 15 seconds, respectively. For 20 epochs, the CNN technique's execution time increased by 8.95 seconds. In the end, the developed model executes for 20 epochs in 3.7 seconds. Furthermore, for 100 epochs, the execution times for YOLOv3 and YOLO-v2 increased by 16.8 and 20 seconds, respectively. For 100 epochs, the CNN method's execution time increased by 13.6 s. The developed model eventually reaches an execution time of 6s for 100 epochs. When compared to alternative ways, the planned strategy has achieved a lower execution time. Additionally, it demonstrates how the designed model's computational complexity has increased.

Discussions

The novel method enhanced the performance outcomes of F-score, recall, accuracy, precision, and execution time. Additionally, compared to previous algorithms, less time is needed to find and track human beings in the gathered dataset. However, because the created model runs faster, it increases computational complexity. Eventually, several data sizes are used to validate the performance of the suggested model. **Table 1** contains information about the model's outcomes.

Table 1. Overall Performance

Data sizes (kb)	Performance assessments				
	Accuracy (%)	Precision (%)	Recall (%)	F-score (%)	Execution time (s)
100	99.34	99.45	99.15	99	3.7
200	99.44	99.53	99.22	99.09	4
300	99.53	99.66	99.42	99.23	5.2
400	99.68	99.79	99.55	99.44	5.9
500	99.75	99.89	99.69	99.57	6

Additionally, crow search fitness has been improved by the FR-CNN-CSO model, which tracks humans accurately and effectively. Additionally, training uses 70% of the datasets, whilst testing uses 30%. Consequently, the developed model achieved improved human identification and tracking performance, achieving 99.34%, 99.44%, 99.53%, 99.68%, and 99.75% accuracy, 99.45%, 99.53%, 99.66%, 99.79%, and 99.89% precision, 99.15%, 99.22%, 99.42%, 99.55%, and 99.69% recall, 99%, 99.09%, 99.23%, 99.44%, and 99.57% F-score, and 3.7s, 4s, 5.2s, 5.9s, and 6s execution time for 100, 200, 300, 400, and 500 kb data volumes. **Fig 11** illustrates the accuracy vs. loss performance gain.

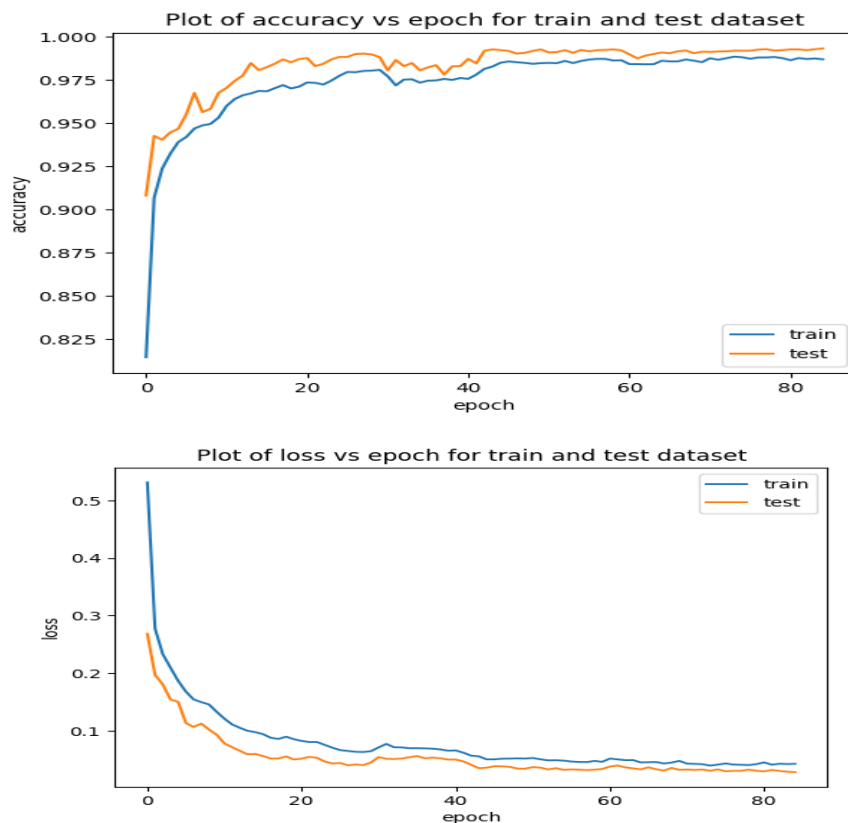


Fig 11. Accuracy Vs. Loss.

As a result, the created model yields better predictions for human tracking systems. Furthermore, the developed model enhances the HTS and reduces computing complexity using an efficient deep learning model.

VI. CONCLUSION

To increase computing efficiency for human tracking systems, a Faster Region-based CNN using the CSO method is developed in this work. In a real-time system, when computational time is a significant parameter that impacts the system's efficiency, optimizing-based deep learning, and one-class detection a difficult challenge. Because the created method takes less time to compute, it offers the advantage of high accuracy. Preprocessing, feature extraction, plus human detection is all included in the suggested approach. 99.34% is the efficiency attained for human detection. Based on many studies, the recall achieved for human target tracking is 99.15%. The outcomes demonstrated that more accurate results can be obtained in a reasonable amount of processing time by employing an FR-CNN-CSO model. It is quite possible to apply such an approach to a complex system as a drone. In the future, hybrid optimization using improved DL models will increase computational complexity and boost human tracking system performance.

CRedit Author Statement

The authors confirm contribution to the paper as follows:

Conceptualization: Srilatha M and Srinivasu N; **Methodology:** Srinivasu N; **Writing- Original Draft Preparation:** Srilatha M and Srinivasu N; **Supervision:** Srinivasu N; **Validation:** Srilatha M; **Writing- Reviewing and Editing:** Srilatha M and Srinivasu N; All authors reviewed the results and approved the final version of the manuscript.

Data Availability

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Conflicts of Interests

The author(s) declare(s) that they have no conflicts of interest.

Funding

No funding agency is associated with this research.

Competing Interests

There are no competing interests.

References

- [1]. M. S. Rohei, E. Salwana, N. B. A. K. Shah, and A. S. Kakar, "Design and Testing of an Epidermal RFID Mechanism in a Smart Indoor Human Tracking System," *IEEE Sensors Journal*, vol. 21, no. 4, pp. 5476–5486, Feb. 2021, doi: 10.1109/jsen.2020.3036233.
- [2]. V.-T. Nguyen, A.-T. Nguyen, V.-T. Nguyen, and H.-A. Bui, "A Real-Time Human Tracking System Using Convolutional Neural Network and Particle Filter," *Intelligent Systems and Networks*, pp. 411–417, 2021, doi: 10.1007/978-981-16-2094-2_50.
- [3]. K. Xue et al., "Robotic seam tracking system based on vision sensing and human-machine interaction for multi-pass MAG welding," *Journal of Manufacturing Processes*, vol. 63, pp. 48–59, Mar. 2021, doi: 10.1016/j.jmapro.2020.02.026.
- [4]. A. Motroni et al., "An RFID Tracking System for Agricultural Safety," 2021 IEEE International Conference on RFID Technology and Applications (RFID-TA), pp. 28–31, Oct. 2021, doi: 10.1109/rfid-ta53372.2021.9617356.
- [5]. O. T. Cetinkaya, S. Sandal, E. Bostanci, M. S. Guzel, M. Osmanoglu, and N. Kanwal, "A Fuzzy Rule Based Visual Human Tracking System for Drones," 2019 4th International Conference on Computer Science and Engineering (UBMK), pp. 1–6, Sep. 2019, doi: 10.1109/ubmk.2019.8907104.
- [6]. J. Xu et al., "Vitality-Enhanced Dual-Modal Tracking System Reveals the Dynamic Fate of Mesenchymal Stem Cells for Stroke Therapy (Small 47/2022)," *Small*, vol. 18, no. 47, Nov. 2022, doi: 10.1002/sml.202270256.
- [7]. M. Revilla-León, J. M. Zeidler, and J. C. Kois, "Digital maxillomandibular relationship and mandibular motion recording by using an optical jaw tracking system to acquire a dynamic virtual patient," *The Journal of Prosthetic Dentistry*, vol. 132, no. 1, pp. 14–19, Jul. 2024, doi: 10.1016/j.prosdent.2022.05.012.
- [8]. L.-P. Tian, L.-Q. Chen, Z.-M. Xu, and Z. (David) Chen, "Wits: An Efficient Wi-Fi Based Indoor Positioning and Tracking System," *Remote Sensing*, vol. 14, no. 1, p. 19, Dec. 2021, doi: 10.3390/rs14010019.
- [9]. J. Azimjonov and A. Özmen, "A real-time vehicle detection and a novel vehicle tracking systems for estimating and monitoring traffic flow on highways," *Advanced Engineering Informatics*, vol. 50, p. 101393, Oct. 2021, doi: 10.1016/j.aei.2021.101393.
- [10]. J. Antonanzas, M. Arbeloa-Ibero, and J. C. Quinn, "Comparative life cycle assessment of fixed and single axis tracking systems for photovoltaics," *Journal of Cleaner Production*, vol. 240, p. 118016, Dec. 2019, doi: 10.1016/j.jclepro.2019.118016.
- [11]. T. Hempel and A. Al-Hamadi, "SLAM-Based Multistate Tracking System for Mobile Human-Robot Interaction," *Image Analysis and Recognition*, pp. 368–376, 2020, doi: 10.1007/978-3-030-50347-5_32.
- [12]. Y. Zou, T. Chen, X. Chen, and J. Li, "Robotic seam tracking system combining convolution filter and deep reinforcement learning," *Mechanical Systems and Signal Processing*, vol. 165, p. 108372, Feb. 2022, doi: 10.1016/j.ymssp.2021.108372.
- [13]. M. Liu, Y. Li, and H. Liu, "3D Gaze Estimation for Head-Mounted Eye Tracking System With Auto-Calibration Method," *IEEE Access*, vol. 8, pp. 104207–104215, 2020, doi: 10.1109/access.2020.2999633.
- [14]. O. Younis, W. Al-Nuaimy, M. H., and F. Rowe, "A Hazard Detection and Tracking System for People with Peripheral Vision Loss using Smart Glasses and Augmented Reality," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 2, 2019, doi: 10.14569/ijacsa.2019.0100201.
- [15]. F. Yang, R. Zhang, and Y. Zhao, "Research on a Visual Sensing and Tracking System for Distance Education," *International Journal of Emerging Technologies in Learning (IJET)*, vol. 14, no. 08, p. 181, Apr. 2019, doi: 10.3991/ijet.v14i08.9827.
- [16]. J. Xie, E. Stensrud, and T. Skramstad, "Detection-Based Object Tracking Applied to Remote Ship Inspection," *Sensors*, vol. 21, no. 3, p. 761, Jan. 2021, doi: 10.3390/s21030761.
- [17]. S. K. Teoh, V. V. Yap, C. S. Soh, and P. Sebastian, "Implementation and optimization of human tracking system using ARM embedded platform," 2012 4th International Conference on Intelligent and Advanced Systems (ICIAS2012), pp. 353–356, Jun. 2012, doi: 10.1109/icias.2012.6306217.
- [18]. Y. Liu, Y. Li, Z. Zhuang, and T. Song, "Improvement of Robot Accuracy with an Optical Tracking System," *Sensors*, vol. 20, no. 21, p. 6341, Nov. 2020, doi: 10.3390/s20216341.
- [19]. P. Zhao et al., "Human tracking and identification through a millimeter wave radar," *Ad Hoc Networks*, vol. 116, p. 102475, May 2021, doi: 10.1016/j.adhoc.2021.102475.
- [20]. A. Larumbe-Bergera, G. Garde, S. Porta, R. Cabeza, and A. Villanueva, "Accurate Pupil Center Detection in Off-the-Shelf Eye Tracking Systems Using Convolutional Neural Networks," *Sensors*, vol. 21, no. 20, p. 6847, Oct. 2021, doi: 10.3390/s21206847.
- [21]. P. Pandiyan, R. Thangaraj, M. Subramanian, R. Rahul, M. Nishanth, and I. Palanisamy, "Real-time monitoring of social distancing with person marking and tracking system using YOLO V3 model," *International Journal of Sensor Networks*, vol. 38, no. 3, p. 154, 2022, doi: 10.1504/ijnsnet.2022.121700.
- [22]. Q. Hao, Z. Wang, and L. Qin, "Design of Beidou Satellite System in Ocean Logistics Real-Time Tracking System," *Journal of Coastal Research*, vol. 94, no. sp1, p. 204, Sep. 2019, doi: 10.2112/si94-043.1.
- [23]. C. Liu and T. Szirányi, "Real-Time Human Detection and Gesture Recognition for On-Board UAV Rescue," *Sensors*, vol. 21, no. 6, p. 2180, Mar. 2021, doi: 10.3390/s21062180.
- [24]. F. Shu and J. Shu, "An eight-camera fall detection system using human fall pattern recognition via machine learning by a low-cost android box," *Scientific Reports*, vol. 11, no. 1, Jan. 2021, doi: 10.1038/s41598-021-81115-9.
- [25]. Zhang, Kevin, et al. "Double anchor R-CNN for human detection in a crowd." *arXiv preprint arXiv:1909.09998* (2019).
- [26]. M.-C. Chiu, H.-Y. Tsai, and J.-E. Chiu, "A novel directional object detection method for piled objects using a hybrid region-based convolutional neural network," *Advanced Engineering Informatics*, vol. 51, p. 101448, Jan. 2022, doi: 10.1016/j.aei.2021.101448.
- [27]. I. Ahmed, M. Anisetti, and G. Jeon, "An IoT-based human detection system for complex industrial environment with deep learning architectures and transfer learning," *International Journal of Intelligent Systems*, vol. 37, no. 12, pp. 10249–10267, May 2021, doi: 10.1002/int.22472.
- [28]. K. Boudjit and N. Ramzan, "Human detection based on deep learning YOLO-v2 for real-time UAV applications," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 34, no. 3, pp. 527–544, Apr. 2021, doi: 10.1080/0952813x.2021.1907793.
- [29]. H. Goyal, K. Sidana, C. Singh, A. Jain, and S. Jindal, "A real time face mask detection system using convolutional neural network," *Multimedia Tools and Applications*, vol. 81, no. 11, pp. 14999–15015, Feb. 2022, doi: 10.1007/s11042-022-12166-x.
- [30]. A. A.N. and S. A.P., "Lightweight and computationally faster Hypermetric Convolutional Neural Network for small size object detection," *Image and Vision Computing*, vol. 119, p. 104396, Mar. 2022, doi: 10.1016/j.imavis.2022.104396.

- [31]. Y. Yamazaki, M. Tamaki, C. Premachandra, C. J. Perera, S. Sumathipala, and B. H. Sudantha, "Victim Detection Using UAV with On-board Voice Recognition System," 2019 Third IEEE International Conference on Robotic Computing (IRC), pp. 555–559, Feb. 2019, doi: 10.1109/irc.2019.00114.
- [32]. H. Chander et al., "Wearable Stretch Sensors for Human Movement Monitoring and Fall Detection in Ergonomics," International Journal of Environmental Research and Public Health, vol. 17, no. 10, p. 3554, May 2020, doi: 10.3390/ijerph17103554.
- [33]. A. G. Hussien et al., "Crow Search Algorithm: Theory, Recent Advances, and Applications," IEEE Access, vol. 8, pp. 173548–173565, 2020, doi: 10.1109/access.2020.3024108.
- [34]. Matsumura, Ryo, and Akitoshi Hanazawa. "Human detection using color contrast-based histograms of oriented gradients." International Journal of Innovative Computing, Information and Control 15.4, 2019.
- [35]. S. Singh, U. Ahuja, M. Kumar, K. Kumar, and M. Sachdeva, "Face mask detection using YOLOv3 and faster R-CNN models: COVID-19 environment," Multimedia Tools and Applications, vol. 80, no. 13, pp. 19753–19768, Mar. 2021, doi: 10.1007/s11042-021-10711-8.