Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2023.029323
Article

Tech Science Press

Robust Deep Transfer Learning Based Object Detection and Tracking Approach

C. Narmadha[1], T. Kavitha[2], R. Poonguzhali[2], V. Hamsadhwani[3], Ranjan walia[4], Monia[5] and B. Jegajothi[6,*]

[1]Electronics and Communication Engineering, Periyar Maniammai Institute of Science & Technology(PMIST), Thanjavur, 613403, India
[2]Department of Computer Science and Engineering, Periyar Maniammai Institute of Science & Technology, Thanjavur, 613403, India
[3]Department of Electrical and Electronics Engineering, Periyar Maniammai Institute of Science and Technology, Thanjavur, 613403, India
[4]Electrical Engineering Department, Model Institute of Engineering & Technology (Autonomous), Jammu, 181122, India
[5]Department of Computer Science and Engineering, Shri Mata Vaishno Devi University, Katra, 182320, India
[6]Center for Energy Research, Chennai Institute of Technology, Chennai, 600069, India
*Corresponding Author: B. Jegajothi. Email: jegajothib@citchennai.net
Received: 01 March 2022; Accepted: 25 May 2022

Abstract: At present days, object detection and tracking concepts have gained more importance among researchers and business people. Presently, deep learning (DL) approaches have been used for object tracking as it increases the performance and speed of the tracking process. This paper presents a novel robust DL based object detection and tracking algorithm using Automated Image Annotation with ResNet based Faster regional convolutional neural network (R-CNN) named (AIA-FRCNN) model. The AIA-RFRCNN method performs image annotation using a Discriminative Correlation Filter (DCF) with Channel and Spatial Reliability tracker (CSR) called DCF-CSRT model. The AIA-RFRCNN model makes use of Faster RCNN as an object detector and tracker, which involves region proposal network (RPN) and Fast R-CNN. The RPN is a full convolution network that concurrently predicts the bounding box and score of different objects. The RPN is a trained model used for the generation of the high-quality region proposals, which are utilized by Fast R-CNN for detection process. Besides, Residual Network (ResNet 101) model is used as a shared convolutional neural network (CNN) for the generation of feature maps. The performance of the ResNet 101 model is further improved by the use of Adam optimizer, which tunes the hyperparameters namely learning rate, batch size, momentum, and weight decay. Finally, softmax layer is applied to classify the images. The performance of the AIA-RFRCNN method has been assessed using a benchmark dataset and a detailed comparative analysis of the results takes place. The outcome of the experiments indicated the superior characteristics of the AIA-RFRCNN model under diverse aspects.

Keywords: Object detection; tracking; deep learning; deep transfer learning; image annotation

1 Introduction

Information technology developments raise the demand for supervision to avoid stealing and leakage of data [1]. So, knowledge-based computational techniques, like tracing and detecting techniques of movable objects, have become as main techniques in surveillance associated with security domain. Especially, the techniques to find the object location or region of interest (RoI) that exist in surveillance images is a major challenge in the area of computer vision related research [2]. Visual Object Tracking (VOT) is a procedure applied to locate the random targeted objects, which is represented by RoI in the video frames. Tracking process finds its applicability in different domains augmented reality, robotics, cinematography, security and surveillance, and even entertainment [3].

The details of object position find helpful when required to deduce high level details and are utilized to minimize the operations. The identification of object location from an input image could be attained by 2 processes [4]. The former is object recognition and the latter is object tracking. In first case, feature extraction takes place on the image and discovers the type relevant to the class of the object by the earlier information. After providing the image, the object could be recognized via the learned approach. Hence, a machine learning (ML) technique is mostly utilized for recognition. However, in the next instance, the pixel details that exist in the RoI will be searched instead of searching the object classes, and the area holding the maximum resemblance is searched for the recently provided input image frames. Thus, object detection relates to the process of discovering a formerly identified object in the applied input image whereas object tracking relates to the task of searching for an object through morphological relationships among nearby frames in the video [5]. Although, the objects that exist in the synthetic or real time images can be tracked under varying image quality, resolution, backdrop, and so on [6]. Thus, the abovementioned traditional tracking methods may not guarantee a higher detection rate in every scenario.

In recent times, several works have been carried out in the process of merging object recognition and tracking models. This technique is named tracking by detection. Different machine learning (ML) techniques are utilized for learning the detector in the identification of objects. The challenges that exist in the object detection process are changes in illumination, object shapes, unexpected movement of objects, etc. To manage these features, real time object recognition and tracking and learning models that execute seamless updating are examined regularly. Convolutional neural network (CNN) models are familiar among different computer vision processes, namely semantic segmentation, object detection, and image classifiers [7]. The development of computer vision (CV) could be applied to the semantic equivalent image they take from visual details. It depends either on the parameter optimization in online or off-line learning to generate discriminating features while managing maximum tracking rate on graphical processing unit (GPU). For resolving the issues forced by modernized applications, like those on embedded systems and robotics [8], tracker should increase the tradeoff between speed and accuracy trade-off to its limit. Modern neural design in CNN-based trackers keeps layers count almost small. Learning a discriminated technique is a subject of practicing the network fine offline besides updating the network and the technique online. The huge amount of video annotation databases like ImageNet VID [9], or TrackingNet [10] assist the procedure of the tracking procedure.

A lot of new trackers have successfully manipulated the dynamic representations obtained by CNN. Among the earlier studies that use CNN are GOTURN [11] and SiamFC [12], which employ shallow networks and execute in real time platforms. GOTURN employed CNN for extracting features from the targeted and searching regions. The features from these regions are then merged utilizing fully connected (FC) layer and evaluated. The network undergoes training an end-to-end offline, for regression, i.e., for predicting the position and object size in the searching area. Abundant data and tremendous augmentation are mandatory for instructing the tracking model efficiently. SiamFC used convolution features, but rather than utilizing the FC layers for comparing the object and search area methods, it performs cross correlation of the features equivalent to the previous one from the final layer. The network is fully

convoluted and undergoes training for classifying process, i.e., it discriminates among the features equivalent to the object and those parallel to the backdrop. The objects location in the search area is selected as the location across high cross-correlation score.

Nowadays, anchor-based ROI selection is integrated into a Siamese technique to track objects, named SiamRPN [13]. The major advantage of utilizing anchors is the matching of bounding boxes of the object is that the tracking technique could manage aspect ratio change when Conventional tracking models usually deal only with size modifications during the maintenance of stable aspect ratio. Another advantage of anchor is the requirement of lower data augmentation in contrary to a bounding box regression technique like GOTURN. It is due to the fact that each anchor is estimated for each potential location and fine tuned for fitting with the ground truth bounding box. Deeper, slower, and new composite trackers using CNN are developed, such as ADNet [14] or MDNet [15], CREST [16]. They perform the tracking process by initially understanding an extremely discriminative technique of the object and then adapting the technique to all frames online. The online adaptation of the object technique incurs more delay for the trackers even on modernized GPU.

This study introduces a robust DL based object detection and tracking approach using Automated Image Annotation with ResNet based Faster RCNN (AIA-FRCNN) model. The AIA-RFRCNN method involves an effective image annotation using Discriminative Correlation Filter (DCF) with Channel and Spatial Reliability tracker (CSR) called DCF-CSRT model. The AIA-RFRCNN model uses Faster RCNN as an object detector and tracker, which involves region proposal network (RPN) and Fast R-CNN. Besides, Residual Network (ResNet 101) model is used as a feature extractor for the generation of feature maps. The performance of the ResNet 101 model is further improved by the use of Adam optimizer for setting the hyper-parameters. Finally, softmax layer is used for classification purposes. A detailed set of experimental analyses takes place to ensure the effective performance of the AIA-RFRCNN model and the results are examined in several aspects.

2 The Proposed AIA-RFRCNN Model

The proposed AIA-RFRCNN model undergoes two major stages for detecting and tracking the objects that exist in the series of frames, namely DCS-CSRT based AIA and Faster RCNN based model. The detailed workflows of these models are provided in the following subsections.

2.1 DCS-CSRT Model

Initially, the input videos are converted into a sequence of frames and the objects in the frame are marked as objects for the creation of record files. The automated DCS-CSRT model is applied as an image annotation tool for annotating the objects present in the input frame. It enables to annotation of the objects in a single frame and makes automated annotation of the objects in all the frames that exist in the video. The localization as well as enhancing procedures are given as follows: The features obtained from the search area are placed at the desired location in the previous time-step and combined with well-known filter h_{z-1}. Here, object undergoes localization by adding the correlation responses weighted by the evaluated channel reliability values w_{z-1}. Hence, a scale is calculated with the help of single scale-space correlation filter. Per-channel filter responses were employed for computing the prediction reliability scores

Here, training area is placed at the desired position which is determined at the localization phase. The forefront and backdrop histograms $\tilde{c}$ were obtained and improved by an exponential moving average with learning value of η_c. The forefront histogram has been acquired using Epanechnikov kernel inside the calculated object's bounding boxes while backdrop is filtered from nearby area which is two times the size of the object. The spatial reliability map m is developed and best filters $\tilde{h}$ have been estimated via

optimization. The per-channel learning reliability weight is determined as $\tilde{w}^{(lern)} = \left[\tilde{w}_1^{(lern)}, \ldots, \tilde{w}_{N_c}^{(lern)}\right]^Z$ are evaluated from the correlation responses.

2.2 Faster R-CNN Based Object Detection and Tracking

Faster R-CNN is generally defined as a detection pipeline, which makes use of region proposal network (RPN) as a region proposal model, and Fast R-CNN as a detector. It is a considerably an enhanced model of R-CNN to attain faster and precise outcomes. It utilizes CNN for generating the object proposals called RPN since classical RCNN and Fast RCNN make use of Selective Search in the initial level, known as RPN. In other words, in Faster RCNN, the RPN initially uses a base network (i.e., ResNet-101) as a feature extractor to generate feature map from the image. Then, it divides the feature map into many squared tiles and slides a small network across every tile in succession. The small network allocates a collection of object confidence scores and coordinates points of bounding boxes in every position of the tile. In this paper, ResNet 101 is used as a feature extractor in RPN, followed by Fast RCNN based object detector.

Residual Networks (ResNet) applies the residual block to solve both degradation and gradient vanishing issues that often arouses in CNN [17]. The residual block is used for enhancing the network efficiency and performance. In specific, ResNet systems accomplished better achievements and stand place in the ILSVRC 2015 classification contest. At the same time, residual block in ResNet executes the remaining ones by including the input and output of the residual block. There are 2 kinds of residual connections as given in the following. The similarity shortcuts (x), could be employed directly while the dimensions are same for both input and output. The residual function is expressed as:

$$y = F(x, W) + x \tag{1}$$

If there is a change in dimension, the shortcuts process the identity mapping along with additional zero entries that are embedded with enhanced dimensions, later the projection shortcut has been applied for matching the dimension with the help of given expression,

$$y = F(x, W) + W_s x \tag{2}$$

where x and W define the input and weight of a residual block; whereas y implies the outcome of a residual block. The fundamental infrastructure namely, Residual Block, 2 and 3 layers deep are projected in Fig. 1.

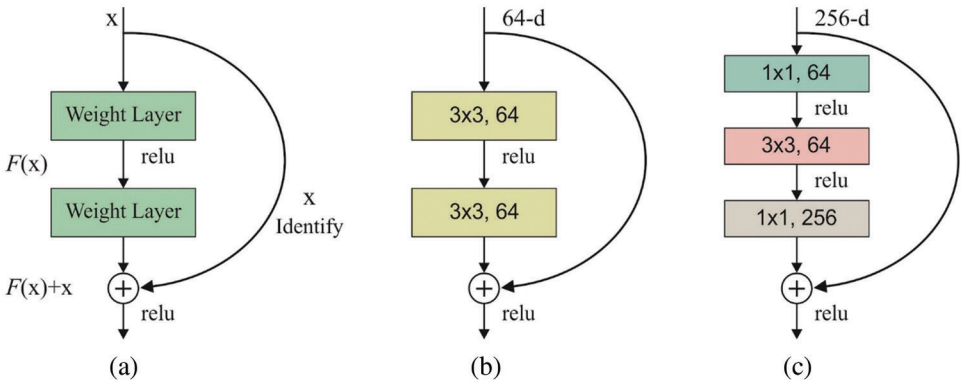

Figure 1: (a) Residual learning building block (b) ResNet two layer block (c) ResNet three layer block

Adam optimizer is an Adaptive Moment estimation optimizer [18], a kind of DL model and it increases the performance at a faster rate. It makes use of a first-order gradient-based optimization technique depending upon the adaptive estimation of low-order moments. The process involved in the Adam optimizer is provided below.

Parameters: α: Learning Rate, $\beta_1, \beta_2 \in [0, 1]$ (Exponential decay rate for the moment estimate), $f(\Theta)$: Stochastic objective function with parameters Θ, Θ_O Initialization of parameter vector

Assumptions: $m_o \leftarrow 0$ (Initialization of first moment vector) *and* $v_o \leftarrow 0$ (Initialization of second moment vector), $t \leftarrow 0$ (Initialization of time step)

while (Θ_t!converged) then

$t \leftarrow t + 1$

$g_t \leftarrow \nabla_\Theta f_t(\theta_t - 1)$ (Find gradient based on stochastic objective at time step t)

$m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1) g_t$ (Updating biased 1st moment estimation)

$v_t \leftarrow \beta_2 v_{t-1} + (1 - \beta_2) g_t$ (Updating biased second actual moment estimation)

$\hat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ (Calculate bias corrected first moment estimation)

$\hat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ (Calculate bias corrected second raw moment estimation)

$\theta_t \leftarrow \theta_{t-1} - \alpha \hat{m}_t / \left(\sqrt{\hat{v}_t} + \varepsilon \right)$ (Updating variable parameters).

End while

The standard parameter settings of the abovementioned variables are $\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999$ and $\varepsilon = 10^{-8}$. Every operation on vectors is element-wise, and β_1^t and β_2^t,represents β_l and β_2 to the power of t.

Faster R-CNN applies CNN for generating object proposals as opposed to Selective Search. RPN applies image feature map is provided as input and produces group of object proposals, and object confidence score as final output. A small network allocates a collection of object classification scores as well as bounding box coordinates for every object location.

Fig. 2 shows the overall process of Faster-RCNN model. The steps followed in Faster R-CNN are provided below:

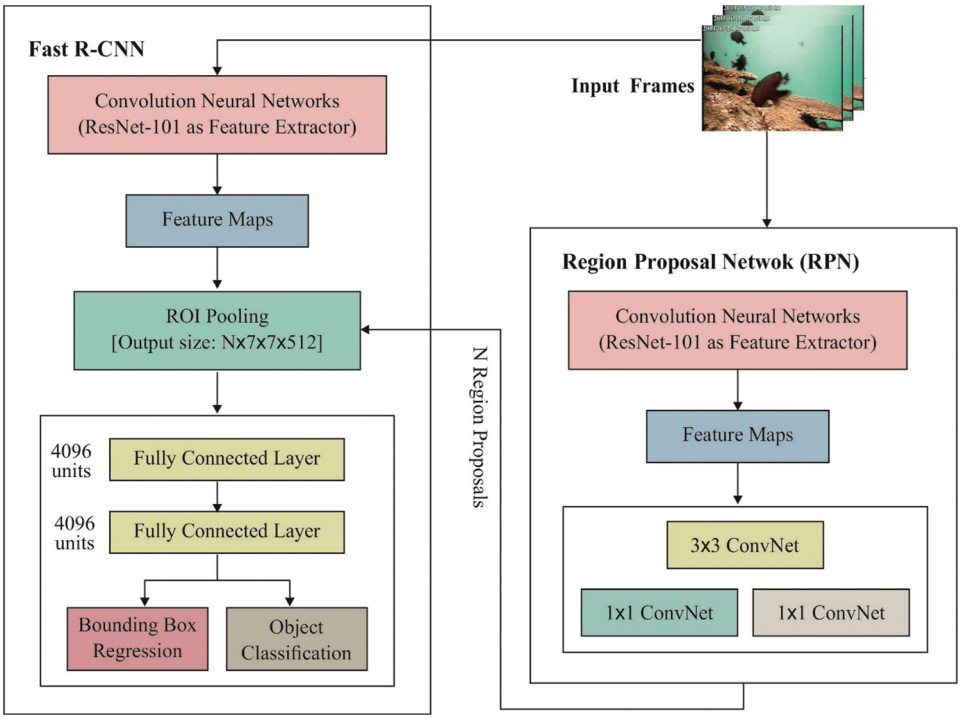

Figure 2: Overall process of faster-RCNN

- Initially, image is provided as an input and it is fed to the ResNet-101 (CNN) finally producing a feature map for concerned image.
- Secondly, RPN is used on feature maps, which provides the object proposals with the objectness score or value.
- Then, RoI pooling layer has been utilized on these proposals which leads to making every proposal a similar size.
- At last, proposals are induced into FC layer which is composed of softmax layer as well as a linear regression layer for classification tasks. Finally, it produces the bounding boxes for every object that exists in the input video frame.

Next 2, layers also apply the same procedure such that 1 × 1 convolutional layer with 18 units is used to classify objects, whereas 1 × 1 convolution with 36 units is consumed for bounding box regressor. Thus, the 18 units in the classifier segment provide a resultant size of (H, W, 18). Subsequently, the results were employed for obtaining the possibilities of having objects in all backbone feature maps inside 9 of the anchors at various points. Therefore, 36 units in the regression segment offer an output size of (H, W, 36), which is then applied to giving 4 regression coefficients for 9 anchors at the backbone feature map. Such regression coefficients were utilized for enhancing the anchor coordinates with objects.

In general, the output feature map is comprised of 40 × 60 positions, such as 40 * 60 * 9~20k anchors. In case of training time, every anchor that exceeds the boundary is removed and it is not responsible for loss of function. It leaves around 6k anchors for an image. Here, an anchor is assumed to be a "positive" instance when it meets the provided constraints namely,

- The anchor should have maximum Intersection over Union (IoU), along with a ground truth box.
- An anchor which has IoU higher than 0.7 with any ground truth boxes. The similar ground truth box leads to various anchors that are allocated to positive samples.
- Then, the anchor is named as "negative" when the IoU with ground truth boxes is minimum than 0.3. Hence, residual (positive or negative) anchors are not capable of RPN training.
- Training loss for RPN is assumed to be a multitask loss, which has been expressed by:

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}\left(p_i, p_i^*\right) + \lambda\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}\left(t_i, t_i^*\right) \tag{3}$$

- In this model, i denotes the index of an anchor in mini-batch. The classification loss $L_{cls}(p_i, p_i^*)$ defines the log loss over 2 classes. p_i implies resultant score from classification branch for anchor i, and p_i^* represents the ground truth label (1 or 0).
- The regression loss $L_{re}(t_i, t_i^*)$ is triggered when the anchor is filled with objects such that the ground truth t_i^* is 1. The term t_i describes the output prediction of regression layer with 4 parameters namely, $t_x, t_y, t_w, and\ t_h$.
- Here, regression coefficients were employed to the anchors that give accurate localization and bounding boxes.
- The attained boxes are organized on the basis of the *cls* values. Thus, non-maximum suppression (NMS) has been used with a threshold of 0.7. First, every bounding box with IoU of a maximum than 0.7 and alternate bounding boxes are not considered for further processing.

The Fast R-CNN has CNN with concerned final pooling layer that is interchanged by an "ROI pooling" layer while final fully connected (FC) layer is replaced by 2 branches such as a (K + 1) category softmax layer branch as well as category-specific bounding box regression branch. The RoI is defined as a neural-net layer that has been applied for object detection process. It was coined by Girshick [19]. It is meant to be an

extensive task employed in object detection tasks with the help of CNN. The key objective is to process max pooling on non-uniform input sizes and acquires fixed-size feature maps (7 × 7), gaining vital enhancement of training and testing. Moreover, maximum detection accuracy can be accomplished. The simulation outcome of ROI pooling layer is the size of (N, 7, 7, 512) where N implies the proposal count from the RPN. When the attained results are passed to 2 FC layers, the features were induced into consecutive classification and regression branches. Here, classification and detection branches vary from RPN. The classification layer is composed of C units for all classes in the detection task. Then, features are provided using softmax layer which leads to the generation of classification values. The regression layer coefficients are utilized for enhancing the detected bounding boxes. The classification layer properly provides the class labels for every detected object that exist in the video frames.

3 Performance Validation

In this section, an extensive series of experiments were carried out on three datasets and the results are analyzed interms of detection accuracy, annotation time, Center Location Error (CLE), and Overlap Rate (OR) [20]. The details of the dataset, measures, and results analysis are discussed in the subsequent subsections. For comparison purposes, region scalable CNN (RS-CNN), Fast R-CNN, multi-object detection and tracking (MDT), MPPCA, and social force (SF) are employed [21].

3.1 Dataset Used

Dataset 1 is a multi-object tracking bird dataset (http://cvlab.hanyang.ac.kr/tracker_benchmark/datasets.html), which consists of 99 frames with the duration of 3 s. The second UCSDped2 (Test004) is an anomaly detection dataset (http://www.svcl.ucsd.edu/projects/anomaly/dataset.htm), which comprises 180 frames and the duration of the video is 6 s. Dataset 3 includes two sub files namely underwater blurred and underwater crowded (https://link.springer.com/article/10.1007/s11554-019-00879-6) which have a total of 2875 and 4600 frames under the duration of 575 s.

3.2 Performance Measures

The set of measures used to determine the performance of the AIA-FRCNN model is detection accuracy, annotation time, CLE, and OR. CLE is mainly used to compute the accurateness and efficiency of the compared tracking models and is represented as follows.

$$CLE = \sqrt{(x' + x)^2 + (y' + y)^2} \tag{4}$$

where (x', y') represents the position of the objects offered by various tracking models, and (x, y) indicates the location of the ground truth values.

The OR indicates the stability level of every tracker since it assumes the size and pose of the target objects. It can be defined by,

$$OR = \frac{RoI_{TBB} \cap RoI_{GTBB}}{RoI_{TBB} \cup RoI_{GTBB}} \tag{5}$$

where RoI_{TBB} indicates the bounding box of the RoI by the proposed method and RoI_{GTBB} denotes the bounding box of the RoI denoted by the ground truth value.

3.3 Results Analysis on Dataset 1

Fig. 3 shows the analysis of the results provided by AIA-FRCNN model and existing models interms of detection accuracy on dataset 1. The figure showed that the SF model has reached the worst detection rate of

the other methods. In continuing, the MDT and MPPCA models have resulted in a slightly higher and near identical detection accuracy. In line with this, the Fast R-CNN model has attained somewhat acceptable detection rate over the other methods. But it failed to show better results over the RS-CNN and AIA-FRCNN models. Concurrently, it is depicted that the near optimal detection accuracy has been provided by the RS-CNN model. However, maximum detection accuracy has been exhibited by the proposed AIA-FRCNN model compared to other models on all the applied frames.

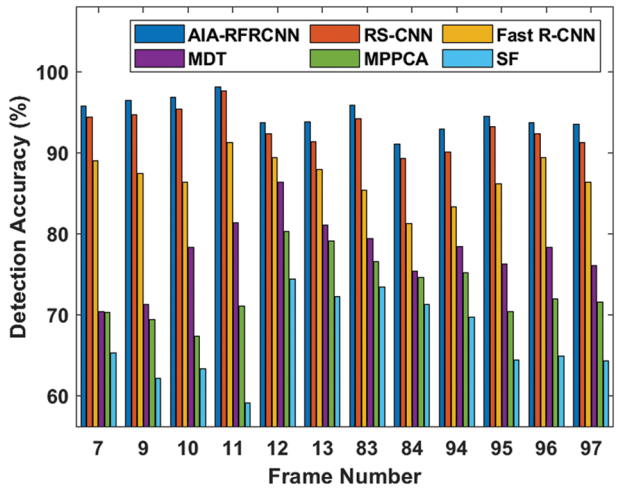

Figure 3: Comparisons of various methods for dataset 1 in terms of detection accuracy

3.4 Results Analysis on Dataset 2

Fig. 4 indicates the analysis of the results provided by AIA-FRCNN model and previous methods with respect to detection accuracy on dataset 2. The figure implied that the SF model has accomplished an inferior detection rate over the alternate approaches. In line with this, the MDT as well as MPPCA methodologies have attained better and closer detection accuracy. Along with that, the Fast R-CNN model has obtained reasonable detection rate over the other methods. Likewise, the RS-CNN model has outperformed and achieved maximum detection rate than the other models. However, it is unsuccessful to imply better results over the IRS-CNN and AIA-FRCNN models. At the same time, it is illustrated that the near optimal detection accuracy has been provided by the IRS-CNN model. Thus, higher detection accuracy has been showcased by the proposed AIA-FRCNN model when compared to other models on all the applied frames.

3.5 Results Analysis on Dataset 3

Fig. 5 showcase the qualitative analysis of the AIA-FRCNN model on dataset 3. The figure depicted that the AIA-FRCNN method has productively detected and tracked multiple objects present in the input video frames in dataset 3. The first column points to the original image and the second column show the corresponding tracked output. The figure demonstrated that the AIA-FRCNN model has predicted all the four objects that exist in the frame and provided a bounding box with detection rate.

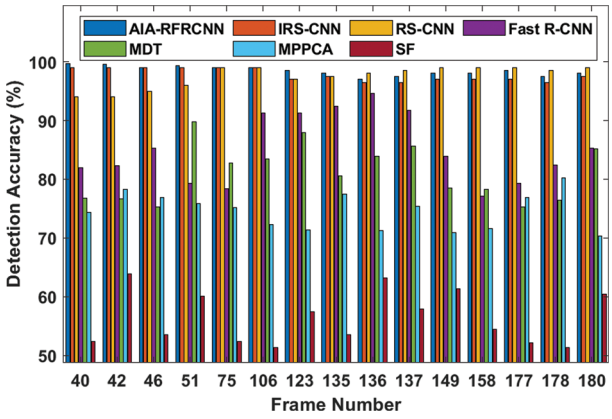

Figure 4: Comparisons of various methods for dataset 2 in terms of detection accuracy

Figure 5: Visualizing objection detection of AIA-RFRCNN for dataset 3 (Blurred)

Fig. 6 exhibits the analysis of the results provided by AIA-FRCNN model and existing models interms of detection accuracy on dataset 3. The figure represented that the SF model has achieved the worst detection rate over the alternate methods. In line with this, the MDT and MPPCA technologies have resulted in a slightly higher and closer identical detection accuracy. Likewise, the Fast R-CNN model has achieved moderate detection rate over the other methods. Hence, it failed to exhibit better results than the RS-CNN and AIA-FRCNN models. At the same time, it is shown that the closer optimal detection accuracy has

been offered by the RS-CNN model. Thus, maximum detection accuracy has been shown by the proposed AIA-FRCNN model compared to other models on all the applied frames.

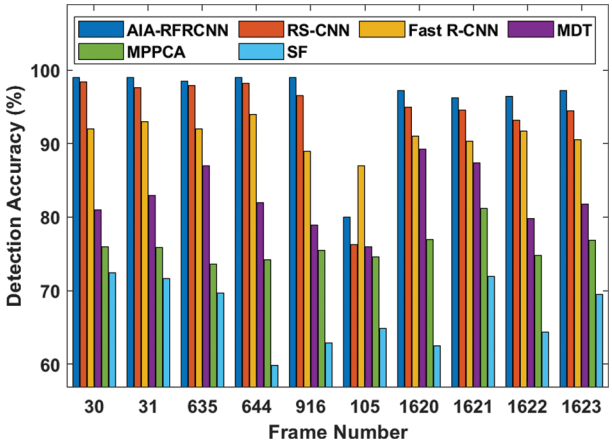

Figure 6: Comparisons of various methods for dataset 3 in terms of detection accuracy

3.6 Analysis of Average CLE

Tab. 1 depicts the results analysis of the AIA-FRCNN and existing models interms of average CLE. On the applied dataset 1, the DSST model has shown its ineffectiveness in tracking the objects and attained a maximum average CLE of 56.72. Next to that, the CF2 and KCF models have achieved somewhat lower CLE over the KCF model by obtaining closer average CLE of 38.50 and 45.57 respectively. Along with that, a moderate average CLE value of 17.38 has been accomplished d by CSK model, which is better than the CLE values provided by the previous models. Moreover, the OMFL and FCT models have resulted in low and nearer CLE values of 7.49 and 9.30 respectively.

Table 1: Results analysis of average CLE (in Pixels) of AIA-RFRCNN with state of art methods

Methods	Dataset 1	Dataset 2	Dataset 3
AIA-RFRCNN	5.67	6.89	4.32
OMFL	7.49	9.21	12.88
CSK	17.38	19.48	29.40
FCT	9.30	15.86	20.79
DSST	56.72	58.31	63.84
CF2	38.50	40.58	48.76
KCF	45.57	47.20	50.42

But the AIA-RFRCNN model has ensured its effective results by attaining a minimum average CLE of 5.67. On the applied dataset 2, the DSST model has shown its ineffectiveness in tracking the objects and attained a higher average CLE of 58.31. Next to that, the CF2 and KCF models have achieved somewhat lower CLE over the KCF model by attaining closer average CLE of 40.58 and 47.20 respectively. Likewise, a moderate average CLE value of 19.48 has been accomplished d by CSK model, which is better than the CLE values provided by the previous models. In addition, the OMFL and FCT models

have resulted in low and nearer CLE values of 9.21 and 15.86 correspondingly. However, the AIA-RFRCNN model has approved effective results by accomplishing a lower average CLE of 6.89.

On the applied dataset 3, the DSST model has showcased its inefficiency in tracking the objects and reached a higher average CLE of 63.84. Followed by, the CF2 and KCF methodologies have attained least CLE over the KCF model by attaining closer average CLE of 48.76 and 50.42 respectively. In line with this, a better average CLE value of 29.40 has been accomplished d by CSK model, which is better than the CLE values provided by the previous models. Furthermore, the OMFL and FCT models have resulted in low and nearer CLE values of 12.88 and 20.79 respectively. But the AIA-RFRCNN model has ensured its effective results by attaining the least average CLE of 4.32.

3.7 Analysis of Results Interms of Overlap Rate

Tab. 2 and Fig. 7 depict the results analysis of the AIA-FRCNN and existing models interms of average CLE. On the applied dataset 1, the DSST model has shown its ineffectiveness in tracking the objects and attained a higher overlap rate of 0.52. Next to that, the CF2 and KCF methodologies have achieved somewhat lower CLE over the KCF model by reaching closer overlap rates of 0.63 and 0.68 respectively. In line with this, a moderate overlap rate value of 0.71 has been attained by CSK model, which is better than the CLE values provided by the existing models. In addition, the OMFL and FCT models have resulted in high and nearer CLE values of 0.78 and 0.74 respectively. But the AIA-RFRCNN model has ensured its effective results by attaining a maximum overlap rate of 0.89.

Table 2: Results analysis of overlap rate of AIA-RFRCNN with state of art methods

Methods	Dataset 1	Dataset 2	Dataset 3
AIA-RFRCNN	0.89	0.86	0.91
OMFL	0.78	0.77	0.72
CSK	0.71	0.68	0.64
FCT	0.74	0.73	0.69
DSST	0.52	0.49	0.46
CF2	0.68	0.65	0.62
KCF	0.63	0.59	0.51

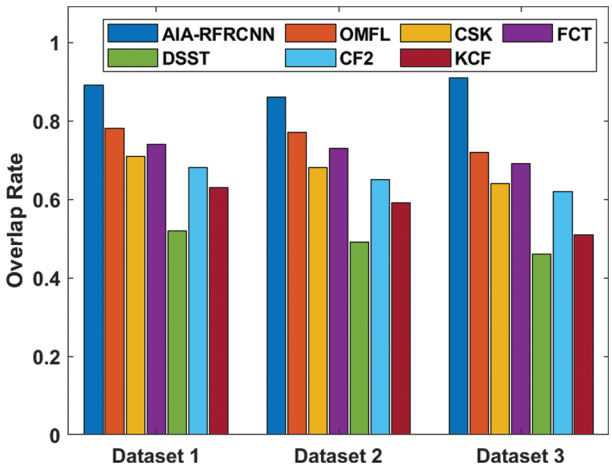

Figure 7: Results analysis of overlap rate of AIA-RFRCNN model

On the applied dataset 2, the DSST model has shown its ineffectiveness in tracking the objects and attained a higher overlap rate of 0.49. Then, the CF2 and KCF models have achieved somewhat lower CLE over the KCF model by obtaining closer overlap rates of 0.65 and 0.59 respectively. Similarly, a moderate overlap rate value of 0.68 has been accomplished d by CSK model, which is better than the CLE values provided by the previous models. Also, the OMFL and FCT models have resulted in high and nearer CLE values of 0.77 and 0.73 respectively. But the AIA-RFRCNN model has ensured its effective results by attaining a maximum overlap rate of 0.86. On the applied dataset 3, the DSST model has shown its inefficiency in tracking the objects and attained a maximum overlap rate of 0.46. Next, the CF2 and KCF models have achieved minimum CLE over the KCF model by obtaining closer overlap rates of 0.62 and 0.51 respectively. Along with that, a moderate overlap rate value of 0.64 has been accomplished by CSK model, which is better than the CLE values provided by the compared models. Moreover, the OMFL and FCT models have resulted in high and nearer CLE values of 0.72 and 0.69 respectively. But the AIA-RFRCNN model has ensured its effective results by attaining a maximum overlap rate of 0.91.

3.8 Result Analysis: Annotation Time

Tab. 3 tabulates the results analysis of the AIA-FRCNN model interms of annotation time. The table values indicated that dataset 1 requires a manual annotation time of 1800 s whereas the proposed automated annotation time of only 1.2 s. Besides, the bird dataset 2 requires a higher manual annotation time of 1200 s which is completely reduced to 0.8 s. Furthermore, the underwater blurred and crowded dataset needs only a minimum of 8.5 and 18.4 s respectively. These reduced annotation times incurred by the presented AIA method indicated that the AIA tool consumes significantly lesser amount of time for annotating the images.

Table 3: Time taken for annotation manual *vs*. AIA tool

Dataset	Testbed	Frames	Manual annotation time (s)	Automated annotation time (s)
Dataset 1	Test004	180	1800	1.2
Dataset 2	Bird	99	1200	0.8
Dataset 3	Blurred	2875	9800	8.5
	Crowded	4600	15600	18.4

From the abovementioned tables and figures, it is evident that the proposed AIA-RFRCNN model is the effective tool for the detection and tracking of objects with maximum detection rate, overlap rate, minimum average CLE and annotation time.

4 Conclusion

This paper has presented a novel robust DL based object detection and tracking algorithm using AIA-RFRCNN model. Initially, the input videos are converted into a sequence of frames and the objects in the frame are marked as objects for the creation of record files. The automated DCS-CSRT model is applied as an image annotation tool for annotating the objects present in the input frame. The AIA-RFRCNN model uses Faster RCNN as an object detector and tracker, which involves RPN and Fast R-CNN. The performance of the ResNet 101 model is further improved by the use of Adam optimizer for setting the hyper-parameters namely learning rate, batch size, momentum, and weight decay. Finally, softmax layer is used for classification purposes. The performance of the AIA-RFRCNN model has been tested against three benchmark video datasets and the results are determined under different measures.

The presented AIA-RFRCNN model has outperformed the existing models with maximum detection accuracy of 94.67%, 98.43%, and 96.15% on the applied dataset 1, 2, and 3 respectively. In the future, hybrid metaheuristics based hyperparameter optimizers can be included to improve the overall performance.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] R. Ravindran, M. J. Santora and M. M. Jamali, "Multi-object detection and tracking, based on DNN, for autonomous vehicles: A review," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 5668–5677, 2021.

[2] S. K. Pal, A. Pramanik, J. Maiti and P. Mitra, "Deep learning in multi-object detection and tracking: State of the art," *Applied Intelligence*, vol. 51, no. 9, pp. 6400–6429, 2021.

[3] D.-H. Lee, "CNN-based single object detection and tracking in videos and its application to drone detection," *Multimedia Tools and Applications*, vol. 80, no. 26–27, pp. 34237–34248, 2021.

[4] A. A. Micheal, K. Vani, S. Sanjeevi and C.-H. Lin, "Object detection and tracking with UAV data using deep learning," *Journal of the Indian Society of Remote Sensing*, vol. 49, no. 3, pp. 463–469, 2021.

[5] F. Bi, X. Ma, W. Chen, W. Fang, H. Chen *et al.,* "Review on video object tracking based on deep learning," *Journal of New Media*, vol. 1, no. 2, pp. 63–74, 2019.

[6] W. Sun, X. Chen, X. R. Zhang, G. Z. Dai, P. S. Chang *et al.,* "A multi-feature learning model with enhanced local attention for vehicle re-identification," *Computers, Materials & Continua*, vol. 69, no. 3, pp. 3549–3561, 2021.

[7] M. J. Gómez-Silva, A. de la Escalera and J. M. Armingol, "Deep learning of appearance affinity for multi-object tracking and re-identification: A comparative view," *Electronics*, vol. 9, no. 11, pp. 1757, 2020.

[8] C.-Y. Tsai and Y.-K. Su, "MobileNet-JDE: A lightweight multi-object tracking model for embedded systems," *Multimedia Tools and Applications*, vol. 81, no. 7, pp. 9915–9937, 2022.

[9] J. Yang, H. Ge, J. Yang, Y. Tong and S. Su, "Online multi-object tracking using multi-function integration and tracking simulation training," *Applied Intelligence*, vol. 52, no. 2, pp. 1268–1288, 2022.

[10] W. Fan, X. Xu, X. Xing, W. Chen and D. Huang, "LSSED: A large-scale dataset and benchmark for speech emotion recognition," in *ICASSP, 2021-2021 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, ON, Canada, pp. 641–645, 2021.

[11] S. T. Sarker and J. M. Banda, "Solar event tracking with deep regression networks: A proof of concept evaluation," in *2019 IEEE Int. Conf. on Big Data (Big Data)*, Los Angeles, CA, USA, pp. 4942–4949, 2019.

[12] H. Huang, G. Liu, Y. Zhang, R. Xiong and S. Zhang, "Ensemble siamese networks for object tracking," *Neural Computing and Applications*, vol. 34, no. 10, pp. 8173–8191, 2022.

[13] B. Li, J. Yan, W. Wu, Z. Zhu and X. Hu, "High performance visual tracking with siamese region proposal network," in *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Salt Lake City, UT, pp. 8971–8980, 2018.

[14] S. Yun, J. Choi, Y. Yoo, K. Yun and J. Y. Choi, "Action-decision networks for visual tracking with deep reinforcement learning," in *2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, pp. 1349–1358, 2017.

[15] L. Zhao, "3D densely connected convolution neural networks for pulmonary parenchyma segmentation from CT images," *Journal of Physics: Conference Series*, vol. 1631, no. 1, pp. 1–4, 2020.

[16] Y. Song, C. Ma, L. Gong, J. Zhang, R. W. H. Lau *et al.,* "Convolutional Residual Learning for Visual Tracking," in *2017 IEEE Int. Conf. on Computer Vision (ICCV)*, Venice, pp. 2574–2583, 2017.

[17] A. Sengupta, Y. Ye, R. Wang, C. Liu and K. Roy, "Going deeper in spiking neural networks: VGG and residual architectures," *Frontiers of Neuroscience*, vol. 13, pp. 1–16, 2019.

[18] D. A. Pustokhin, I. V. Pustokhina, P. N. Dinh, S. V. Phan, G. N. Nguyen *et al.,* "An effective deep residual network based class attention layer with bidirectional LSTM for diagnosis and classification of COVID-19," *Journal of Applied Statistics*, vol. 25, pp. 1–18, 2020.

[19] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE Int. Conf. on Computer Vision*, Santiago, Chile, pp. 1440–1448, 2018.

[20] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[21] M. Y. Abbass, K.-C. Kwon, N. Kim, S. A. Abdelwahab, F. E. Abd El-Samie *et al.,* "Utilization of deep convolutional and handcrafted features for object tracking," *Optik*, vol. 218, pp. 164926, 2020.