
Robust Deep Learning Empowered Real Time Object Detection for Unmanned Aerial Vehicles based Surveillance Applications

C. Prasanna Ranjith¹, Bhalchandra M. Hardas²,
M. Syed Khaja Mohideen³, N. Nijil Raj⁴,
Nismon Rio Robert⁵ and Prakash Mohan^{6,*}

¹*Faculty in Information Technology Department, University of Technology and Applied Sciences, Shinas, Sultanate of Oman*

²*Electronics Engineering Dept, Shri Ramdeobaba College of Engineering and Management, Nagpur, India*

³*Department of Information Technology, University of Technology and Applied Science – Salalah, Sultanate of Oman*

⁴*Department of Computer Science and Engineering Younus College of Engineering and Technology, Kollam, India*

⁵*Department of Computer Science, Christ University, Bangalore, India*

⁶*School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India*

*E-mail: Prasanna.Christodoss@shct.edu.om; hardasbm@rknec.edu;
s.mohideen@sct.edu.om; nijilrajn@ymail.com; nismon.rio@christuniversity.in;
m.prakash@vit.ac.in*

**Corresponding Author*

Received 13 September 2021; Accepted 22 November 2021;
Publication 15 November 2022

Abstract

Surveillance is a major stream of research in the field of Unmanned Aerial Vehicles (UAV), which focuses on the observation of a person, group of people, buildings, infrastructure, etc. With the integration of real time images and video processing approaches such as machine learning, deep

Journal of Mobile Multimedia, Vol. 19_2, 451–476.

doi: 10.13052/jmm1550-4646.1925

© 2022 River Publishers

learning, and computer vision, the UAV possesses several advantages such as enhanced safety, cheap, rapid response, and effective coverage facility. In this aspect, this study designs robust deep learning based real time object detection (RDL-RTOD) technique for UAV surveillance applications. The proposed RDL-RTOD technique encompasses a two-stage process namely object detection and objects classification. For detecting objects, YOLO-v2 with ResNet-152 technique is used and generates a bounding box for every object. In addition, the classification of detected objects takes place using optimal kernel extreme learning machine (OKELM). In addition, fruit fly optimization (FFO) algorithm is applied for tuning the weight parameter of the KELM model and thereby boosts the classification performance. A series of simulations were carried out on the benchmark dataset and the results are examined under various aspects. The experimental results highlighted the supremacy of the RDL-RTOD technique over the recent approaches in terms of several performance measures.

Keywords: Surveillance, deep learning, unmanned aerial vehicles, object detection, computer vision, image processing.

1 Introduction

Recently, autonomous Unmanned Aerial Vehicle (UAV) particularly drones armed with cameras became widespread with the broad range of their applications like search and rescue, surveillance, infrastructural inspection, precision agriculture, aerial mapping, and so on. Understanding visual information gathered from this drone independently is the region of growing interest [1]. Visual object detection is an important aspect of the application of drones and is crucial to have in completely independent system. But, object recognition using drones is highly complex and challenging since it is susceptible to many imaging situations like low resolution, noise in an image, small target size, blur, and so on [2]. Even the tasks are highly complex due to the constrained computation resource accessible on the drone and the need for all realtime performances in various applications like traffic management, navigation so on. Various UAVs research has been conducted for detecting specific applications in individual objects like pedestrians, vehicles, autonomous landing, and navigation, and several researchers identify many objects. Object identification is detection of multiple objects existing within an image [3]. It is widely separated into 2 phases such as localization and classification. It is an additional problem to object classification in this manner that it precisely locate each object existing in an image as well detect their existence. Object

recognition has some significant application in which several tasks requires human management could be automatic by identifying object in an image [4]. General object recognition goals at locating and classifying objects, label with rectangular bounding box indicate the confidence rate of predictions.

Several UAV researches have attempted to track and detect specific kinds of objects like landmarks for autonomous landing and navigation, vehicle and person include moving pedestrian in realtime. But, there are a few which considered multiple object recognition [5]. In spite of the fact that multiple target object recognitions are very significant for several applications of UAVs. From our perspective, the major reason for the gaps among applications needs and technical abilities are because of the realtime key limitation: (1) object detection algorithm should be frequently handtuned for certain types of context and object; (2) it is complex to store and build a range of target object model, particularly if the object is different in its occurrence, and (3) realtime object recognition demand higher computing power to identify individual object, much lesser while multiple target objects are included [6].

But, the initial problem is eroding because of an advanced breakthrough technique in computer vision that works well on a broad range of objects. Almost all this methods are depending on “deep learning (DL)” using Convolution Neural Network (CNN), and have provided striking performances increase on a variety of detection problem [7]. The main concept is to learn the object model from raw pixel information, rather than employing hand-tuned features as in conventional detection approach. Typically, training these deep models require huge training dataset, however, these problems have been conquered through largescale labelled databases such as ImageNet [8]. Unfortunately, this advanced technique requires extraordinary amount of computations; the numbers of parameters in object models are usually in billions/millions, require gigabytes of memory, and recognition and training with an object model need higher end Graphics Processing Unit (GPU). With these advanced methods at lower cost, lightweight drone isn't feasible due to the power, size, and weight requirement of this device.

Aerial vehicles (UAVs) have been added to the ubiquitous network due to the fast adoption of the Internet of Things. For a wide range of IoT applications, UAVs provide a practical answer to existing terrestrial IoT infrastructure's constraints. UAVs will eventually dominate low-altitude airspace above populated cities, if they live up to expectations. With increased traffic, new research issues arise, such as UAV safety management. Recently, autonomous Unmanned Aerial Vehicles (UAVs), namely drones equipped with cameras, have gained popularity due to the breadth of their applications, which include search and rescue, surveillance, infrastructural inspection,

precision agriculture, and aerial mapping. The area of increasing interest is deciphering visual data collected independently from this drone [1]. Visual object detection is critical for drone applications and must be implemented in a completely self-contained system. However, object recognition using drones is extremely complex and tough due to the fact that it is subject to a variety of imaging conditions such as low resolution, image noise, tiny target size, and blur [2]. Even still, the tasks are extremely hard due to the drone's limited computational resources and the requirement for real-time performance in a variety of applications such as traffic control and navigation. Numerous scholars have undertaken study on UAVs for the purpose of identifying specific applications in individual items such as pedestrians, cars, autonomous landing, and navigation, and some researchers have identified numerous objects. The recognition of several objects inside a picture is referred to as object identification [3]. It is frequently divided into two stages: localization and classification. It adds another layer of complexity to object categorization in that it must correctly locate and recognise each object in an image. Object recognition has a number of key applications in which numerous processes that require human management can be automated through the identification of objects in images [4]. The general object recognition objective is to locate and classify items; the label with the rectangular bounding box indicates the prediction's confidence level.

Numerous UAV researchers have attempted to track and recognise specific types of things in real time, such as landmarks for autonomous landing and navigation, as well as vehicles and people, including moving pedestrians. However, a few took multiple object recognition into account [5]. Despite the fact that recognition of multiple targets is critical for a variety of UAV applications. According to our assessment, the primary reason for the disconnect between application requirements and technical capabilities is the real-time key constraint: (1) object detection algorithms must be frequently fine-tuned for specific types of context and object; (2) it is difficult to store and build a range of target object models, especially when the object is unique in its occurrence; and (3) real-time object recognition requires more computing power to identify a single target object, but significantly less when multiple target objects are included [6]. However, the initial issue is disintegrating as a result of a breakthrough technique in computer vision that works effectively with a wide variety of objects. Almost all of these methods are based on "deep learning (DL)" with Convolutional Neural Networks (CNN), and have demonstrated significant performance improvements across a range of detecting problems [7]. The basic idea is to infer the object model from raw

pixel data, rather than using hand-tuned characteristics as is the case with conventional detection. Typically, training these deep models requires a sizable training dataset; however, these challenges have been overcome by the use of big labelled datasets such as ImageNet [8]. Unfortunately, this advanced technique takes an enormous amount of work; object models typically have billions/millions of parameters, require gigabytes of memory, and recognition and training with an object model require a high-end graphics processing unit (GPU). Due to the device's power, size, and weight requirements, using these advanced approaches at a reduced cost is not practical. The limitations are

- An object detection algorithm should be handtuned on a regular basis for specific types of context and object;
- It is difficult to store and build a range of target object models, especially if the object is unique in its occurrence; and
- Realtime object recognition requires more computing power to identify a single target object, but much less when multiple target objects are included.

The rest of paper is organized as follows: Section 2 presents the related works. Section 3 illustrated Real Time Object Detection Model, Object Detection using YOLO-v2 Model and OKELM model. Section 4 presents the validation of the proposed systems and its experiments results. Section 5 outlines the future enhancement of proposed systems and conclusion.

2 Related Works

Subbulakshmi [9] proposed a cost efficient method for aerial observation where they move larger computational tasks to the cloud when retaining constrained computational onboard UAV devices with edge computing method. Furthermore, the presented method would preserve minimal communications among cloud and UAV. Therefore, it minimizes the end to end delay and network traffic. Ambeth Kumar [10] proposed a deep feature pyramid framework that utilizes inherent property of extracted features from Convolution Network by capturing more general features in the images (like color, edge, and so on.) as well as the minute thorough feature to the class contain in this problems. They employ VisDrone18 datasets for the research that has distinct objects like bicycles, pedestrians, vehicles, and so on.

Sambat et al. [11] aim are to perform traffic analyses with UAV based videos and DL methods. The road traffic videos are gathered via position fixed UAV. The advanced DL methods are employed for identifying the

moving object in video. The related mobility metric is estimated for conducting traffic analyses and calculate the consequence of traffic congestion. Nguyen et al. [12] implement and propose a realtime fire recognition solution for wide ranging observation with the UAV using an incorporated alarm system and visual detection. The process consists of a lightweight companion computer, lower cost camera, telemetry, and localization modules along with flight controller. In order to attain realtime recognition, SSD algorithms are performed as a core of the system. Also, they employed MobileNet base models that are very effective for embedded and mobile vision applications.

In Lai and Huang [13], the recognition of a moving fixed wing UAV using DL based distance calculation for conducting research of SAA and midair collision avoidance of UAV is presented with a monocular camera for detecting and tracking a received UAVs. Quadrotors are considered as a possessed UAV, also it is capable of estimating the distance of incoming fixed wing intruders. The adapted object recognition methods are depending on YOLO object detectors. The DNN and CNN models are used for examining their performances in the distance calculation of moving objects. The extracted features of fixed wing UAV is depended on the VGG-16 models, later the results are employed in the distance network for estimating the object distance. We compare the performance of the DNN and CNN models in calculating the distance between moving objects. The retrieved features of the fixed wing UAV are compared to the VGG-16 models, and the findings are then used to estimate the object distance using the distance network. Dong et al. [14] gathered novel thermal image datasets taken through the drones. Next, they employed many distinct DCNN models for training survivor recognition models on these datasets, include YOLOV3-MobileNetV3, YOLOV3, and YOLOV3-MobileNetV1. Because of the constrained computing memory and power of the on-board microcomputer, to balance the accuracy and inference time, they establish an optimum point to finetune and prune the survivor recognition networks on the basis of sensitivity of the convolution layers.

The output of the convolution is then passed through a ReLU activation unit (Rectified Linear Unit). This unit transforms the data into a nonlinear form. Only when the convolution output is negative does the ReLU output become zero.

This paper focuses on the design of robust deep learning based real time object detection (RDL-RTOD) technique for UAV surveillance applications. The proposed RDL-RTOD technique involves YOLO-v2 with ResNet-152 technique for object detection process.

2.1 Object Detection Process

The object detection algorithm divides the surveillance footage into frames using the YOLO-v2 model. [15] YOLO 9000 was introduced by J. Redmon and A. Farhadi in 2016. YOLO v2 object identification deep network includes feature extraction and detection. It uses deep transfer learning to extract features from ResNet-152. It has convolution, transform, and outcome layers. The convolution layer activation is removed, improving DNN control. The transform layer alters the bounding box prediction in summary of target box summaries. the goal's pure bounding box locations.

2.2 Object Classification Process

The OKELM model can classify items based on their detection and bounding box generation. To improve the robustness of extreme learning machines, KELM converts non-separable data into separable data using Mercer's condition to create the kernel matrix. Besides, the classification of detected objects takes place using optimal kernel extreme learning machine (OKELM). Moreover, fruit fly optimization (FFO) algorithm is applied for tuning the weight parameter of the KELM model and thereby boosts the classification performance. A sequence of experimental validation processes takes place on the benchmark dataset and the results are inspected under several aspects.

3 The Proposed Real Time Object Detection Model

In this study, a new RDL-RTOD technique is developed for real time object detection and classification in UAV surveillance applications. The proposed RDL-RTOD technique encompasses a two-stage process namely YOLO-v2 with ResNet-152 based object detection and OKELM based object classification. The FFO algorithm is applied for tuning the weight parameter of the KELM model and thereby boosts the classification performance. Figure 1 demonstrates the overall block diagram of RDL-RTOD model. These two modules are offered in the following sections.

3.1 Object Detection Using YOLO-v2 Model

At the initial stage, the surveillance videos are separated into a sequence of frames and the object detection process is carried out using YOLO-v2 model. In 2016, YOLO v2 and YOLO 9000 has been presented by J. Redmon and A. Farhadi [15]. The YOLO v2 object detection deep network has been

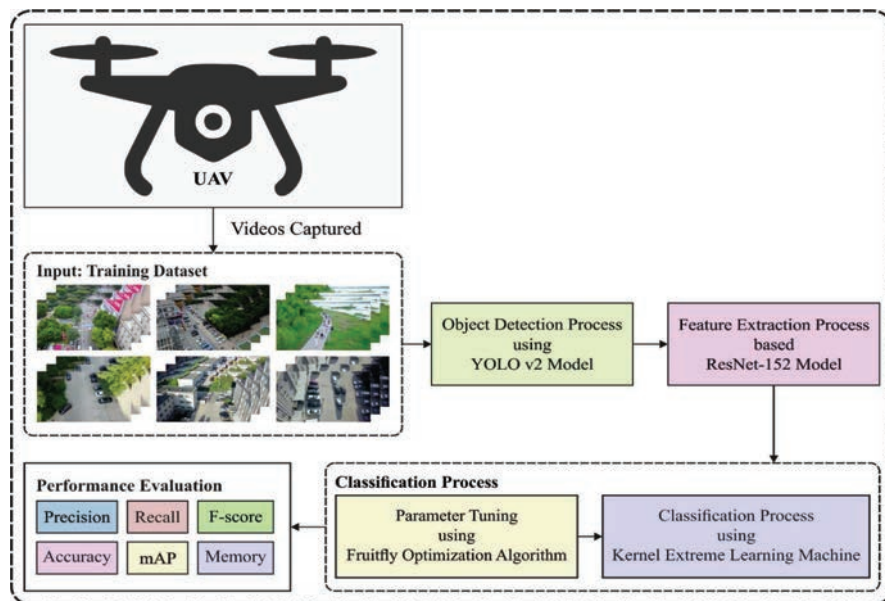


Figure 1 Overall block diagram of RDL-RTOD model.

collected by feature extraction networks and detection networks. The feature extraction networks (ResNet-152) have been deep transfer learning (DTL) technique. During the presented method, ResNet-152 was utilized as deep transfer method to feature extractions. The detection network (YOLO v2) has been CNN involve some convolution, transform, and lastly resultant layers. The transform layer removes activation of convolution layer and enhances the control of DNN. The transform layer changes the bounding box prediction that exists in summaries of target box. The places of pure bounding box of the aim have been generated by the resultant layers.

There should be adequate information about previous study findings. This study develops a more effective real-time object identification and classification technique for unmanned aerial vehicles (UAVs). The proposed RDL-RTOD technique consists of two steps: YOLO-v2 with ResNet-152 based object detection and OKELM based object classification, with the first stage being object detection and the second stage being object classification. This method is used to fine-tune the KELM model's weight parameter, resulting in improved classification accuracy. Readers are encouraged to follow the current study's rationale and procedures.

For computing MSE, loss amongst trained forecasted bounding boxes and target from YOLO v2, the loss function of YOLO v2 has been computed as [16]:

$$YOLOv2loss = Localizationloss + Confidenceloss + Classificationloss \quad (1)$$

The localization loss (LL) measures error amongst the target as well as forecast bounding box. The coefficient to calculate the LL contains the width (w) and height (h) of grid cell of bounding box. The LL coefficients were computed as Equation (2).

$$x_i - \hat{x}_i^2 +$$

$$Localizationloss = q_1 \sum_{a=0}^{g^2} \sum_{b=0}^v 1_{ij}^{obj}$$

$$+ q_1 \sum_{a=0}^{g^2} \sum_{b=0}^v 1_{ij}^{obj} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \quad (2)$$

Where q_1 refers the weight, g implies the amount of grid cells, v represents the amount of bounding boxes from all g , (x_i, y_i) refers the center of v from g , (w_i, h_i) indicates the width and height of v from g , (\hat{x}_i, \hat{y}_i) denotes the center of target from g , (\hat{w}_i, \hat{h}_i) defines the center of target from g , 1_{ij}^{obj} is 1 if there is an object from v in all g otherwise 0.

The confidence loss calculates the confidence score of error if an object has been identified from v bounding box of g . The coefficients of confidence loss are computed in Equation (3).

$$cs_i - \hat{cs}_i^2$$

$$cs_i - \hat{cs}_i^2 + q_3 \sum_{i=0}^{g^2} \sum_{j=0}^v 1_{ij}^{noobj}$$

$$Confidenceloss = q_2 \sum_{a=0}^{g^2} \sum_{b=0}^v 1_{ij}^{obj} \quad (3)$$

Where q_2, q_3 implies the weight of confidence errors, cs_i signifies the confidence score of v from g , \hat{cs}_i refers the confidence score of target from

g , 1_{ij}^{obj} is 1 if there is an object in v in all g then 0, 1_{ij}^{noobj} is 1 if there is no object from v in all g else 0.

The classification loss (CL) measures the error amongst the class conditional probability to all classes from the grid cells i . The parameter to calculate the CL is determined in Equation (4).

$$p_i(z) - \widehat{p}_i(z)^2$$

$$Classificationloss = q_4 \sum_{a=0}^{g^2} 1_i^{obj} \sum_{z \in classes} \quad (4)$$

Where q_4 implies the weight of Classification errors, $p_i(z)$ and $\widehat{p}_i(z)$ implies the probabilities of object evaluated and actual conditional class from grid cells a .

3.2 Object Classification Using OKELM Model

When the objects are effectually detected and bounding boxes are generated, they can be prominently classified using the OKELM model. Kernel extreme learning machine (KELM) improves the robustness of extreme learning machine (ELM) by converting linearly non-separable data in a low-dimensional space to a linearly separable data set... and by employing Mercer's condition to define KELM's kernel matrix. The current study offers a novel learning technique for the kernel extreme learning machine (KELM) that is based on chaotic moth-flame optimization (... Recently, [11] extended the basic ELM theory and proposed the kernel extreme learning machine theory (KELM). The kernel ELM has been established in ELM utilizing the kernel transformation technologies that allow it to optimum generalized efficiency than ELM because of the kernel transforming in input to kernel spaces. Minimize the trained errors and resultant weights simultaneously, KELM is resulting as the subsequent constrained optimized procedure.

$$\min_{\beta} \frac{1}{2} \|\beta\|_2^2 + \frac{C}{2} \sum_{i=1}^n \varepsilon_i^2$$

$$s.t. \phi^T(x_i)\beta = t_i - \varepsilon_i, i = 1, 2, \dots, n \quad (5)$$

where $\phi(\cdot)$ implies the kernel transforming from input to kernel spaces, ε_i refers the i th trained error, the stated parameter C has been utilized for signifying the trade-off amongst β and ε . With appropriate kernel transforming, the linear unsolvable issue from the input space is changed to a

linear solvable issue from the kernel space. There are several single level reduction formulations, but one of the most common is to replace the lower level optimization problem with its Karush-Kuhn-Tucker (KKT) conditions. Such a reduction strategy has been widely used in both classical optimization and evolutionary computation literature. However, KKT conditions contain a set of non-linear equality constraints that are frequently difficult to satisfy. Based on the Karush-Kuhn-Tucker (KKT) proposal and then presenting the Lagrange multiplier α_i , the subsequent dual optimized issue is employed for solving the resultant weights β .

$$\min_{\beta, \alpha, \varepsilon} \left\{ L = \frac{1}{2} \|\beta\|_2^2 + \frac{C}{2} \sum_{i=1}^n \varepsilon_i^2 - \sum_{i=1}^n \alpha_i (\phi^T(x_i) \beta - t_i + \varepsilon_i) \right\} \quad (6)$$

Assume the partial derivative and get them 0, the KKT situations are expressed as:

$$\begin{aligned} \frac{\partial L}{\partial \beta_j} &= 0, j = 1, \dots, L \rightarrow \beta = \Phi^T \alpha \\ \frac{\partial L}{\partial \varepsilon_i} &= 0, i = 1, \dots, n' \rightarrow \alpha = C \varepsilon \\ \frac{\partial L}{\partial \alpha_i} &= 0, i = 1, \dots, n \rightarrow \Phi \beta - t + \varepsilon = 0 \end{aligned} \quad (7)$$

where $\phi(x_1) \cdots \phi(x_n)^T$ refers the kernel output functions [17]. With little bit of simple change and derivation, the resultant function is transforming as to the subsequent expression:

$$\begin{aligned} I_n/C + \Phi \Phi^T &^{-1} t \\ f(x) &= \phi^T(x) \Phi^T \alpha = \phi^T(x) \Phi^T \end{aligned} \quad (8)$$

where I_n refers the identity matrix with n -dimensional, and dependent upon ridge regression theory, the improvement of regulation item I_n/C has been capable of improving the generalized efficiency. To the convenience of computation, kernel transforming was uniformly expressed as inner product, and the kernel matrix has determined as:

$$K = \Phi \Phi^T : K_{i,j} = \phi(x_i) \cdot \phi(x_j) = k(x_i, x_j) \quad (9)$$

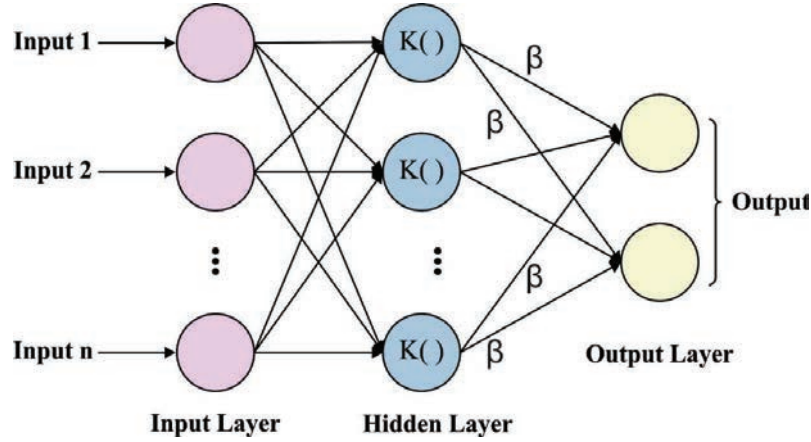


Figure 2 KELM structure.

The network framework of the original KELM technique has demonstrated in Figure 2, containing the input feature layers, the kernel mapping layers, and resultant layers. During the kernel mapping layers, each trained instance is utilized as hidden node. Therefore, the resultant function has been demonstrated in the compact design as:

$$f(x) = [k(x, x_1) \cdots k(x, x_n)]\alpha \quad (10)$$

where $\alpha = \frac{I_n/C + K^{-1}t}{I_n/C + K^{-1}t}$ refers the output weight with respect to kernel mapping. The KELM is same as the SVM in which only requires defining the kernel matrix with no requiring knowledge on the particular kernel mapping and the dimensional of kernel spaces. The KELM assumes structural risk minimized approach for ensuring optimum generalized efficiency with widely regarding empirical risk and confidence interval. The solution of weight α mostly contains the inverse computation of $n \times n$ matrix.

In order to determine the weight parameter of the KELM model, the FFO algorithm is employed. The FFO technique is a novel swarm-based evolutionary technique simulated by the food find performance of FF. Related to another identical species, the FF are higher from vision as well as osphresis. If the food is exposed, it smells the food source from the air utilizing olfactory organs and flies near the way in the beginning; afterward getting nearby the food place, the FF is also utilizing their sharp vision for finding food and another FF flocking place and fly to that way. Based on the food exploring

features of FF swarms, the fundamental FOA is reviewing in seven steps that are explained as:

Step 1. Initialization of the parameter and population place. An essential parameter of FOA is the population size PS , the maximal iteration numbers T , and the arbitrary flight distance Random-Value [18]. A primary place of $-axis$ FF swarms $-axis, Y$ has been distributed arbitrarily from the search space X utilizing Equation (11):

$$\begin{aligned} X_{axis_j} &= LB_j + (UB_j - LB_j) * rand() \\ Y_{axis_j} &= LB_j + (UB_j - LB_j) * rand() \end{aligned} \quad (11)$$

where LB_j and UB_j represents the lower as well as upper bounds of FF swarm place from the j th dimension exploring scope correspondingly, $j = 1, 2, \dots, n$, n refers the dimensional of issues that exist resolved.

Step 2. Create the way and distance of FF separate arbitrarily to the search of food:

$$\begin{aligned} X_{i,j} &= X_{axis_j} + RandomValue \\ -axis_j + RandomValue \\ y_{i,j} &= Y \end{aligned} \quad (12)$$

where $i = 1, 2, \dots, PS$, $X_{i,j}$ signifies the place of i th FF of the j th dimensional from the X -direction.

Step 3. Compute the distance amongst the food place and the origin at primary, afterward the smell concentration judgment was more computed that is the reciprocal of distances:

$$\begin{aligned} Dist_{i,j} &= \sqrt{(X_{i,j}^2 + Y_{i,j}^2)} \\ S_{i,j} &= \frac{1}{Dist_{i,j}} \end{aligned} \quad (13)$$

Step 4. An input the smell concentration judgment as to smell concentration judgment function (or named as Fitness function) for obtaining the smell concentration of FF separate place:

$$Smell_i = Function(S_{i,j}) \quad (14)$$

Step 5. To determine the separate with higher smell concentration (the higher value of Smell) in the FF swarm.

$$[bestSmellbestIndex] = \max(Smell) \quad (15)$$

Step 6. Contrast the present maximal smell concentration value with their historical optimum value (Smellbest), if bestSmell Smellbest, afterward upgrade the higher smell concentration value as well as coordinate. The bestIndex represents the order number of present optimum separate. The FF swarm flies nearby place of an optimum separate with utilizing vision:

$$Smellbest = bestSmell \quad (16)$$

$$X_{axis} = X(bestIndex) \quad (17)$$

$$-axis = Y(bestIndex)$$

Step 7. Repeat the osphresis as well as vision searching phases Steps 2–6, next judge when the smell concentration has maximum to the preceding iterations of smell concentration or iterative number attains the maximal iteration count, then, end the loop.

4 Performance Validation

In order to validate the results, the VisDrone [19] dataset is employed which has different aerial images. It has 263 videos designed by 179,264 frames and 10,209 static images and contains various objects namely pedestrians, vehicles, bicycles, etc., and density (sparse and crowded scenes). Figure 3 depicts a few sample images.

Table 1 and Figure 4 offer the overall results analysis of the RDL-RTOD technique on test VisDrone dataset. The results demonstrated that the RDL-RTOD technique has gained maximum performance with an accuracy of 95.87%, precision of 90.57%, recall of 96.14%, and F-score of 92.12%. Table 1 and Figure 4 summaries the RDL-RTOD technique's overall performance on a test VisDrone dataset. The results indicated that the RDL-RTOD technique performed optimally, with an accuracy of 95.87 percent, a precision of 90.57 percent, a recall of 96.14 percent, and an F-score of 92.12 percent.

A brief comparative accuracy analysis of the RDL-RTOD technique with existing ones takes place in Table 2 and Figure 5. The results depicted that the



Figure 3 Sample images.

Table 1 Results analysis of RDL-RTOD technique

Measures	Values (%)
Accuracy	95.87
Precision	90.57
Recall	96.14
F-Score	92.12

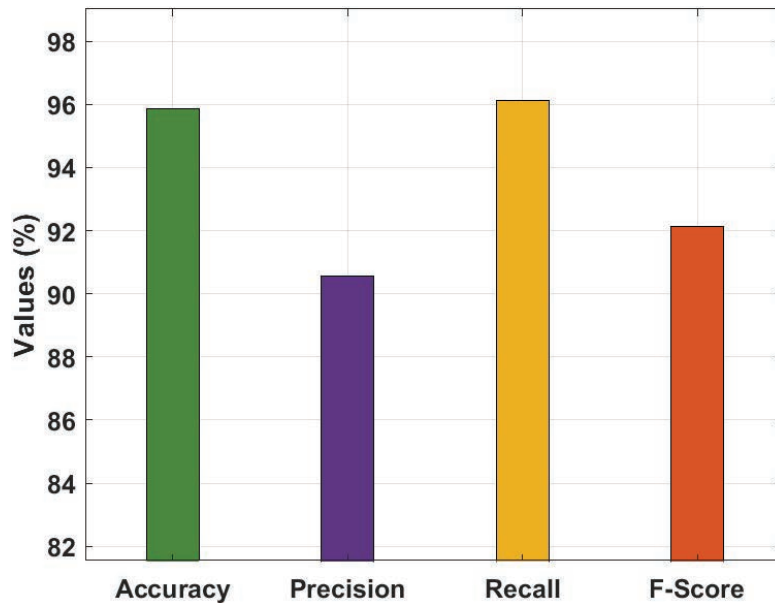


Figure 4 Result analysis of RDL-RTOD model.

Table 2 Comparative accuracy analysis of RDL-RTOD technique

Methods	Accuracy (%)
SSD-MobileNet	71.00
SSD-ResNet50	76.00
SSD-GoogleNet	79.00
FRCNN- MobileNet	70.50
FRCNN ResNet50	76.00
FRCNN-GoogleNet	79.00
RDL-RTOD	95.87

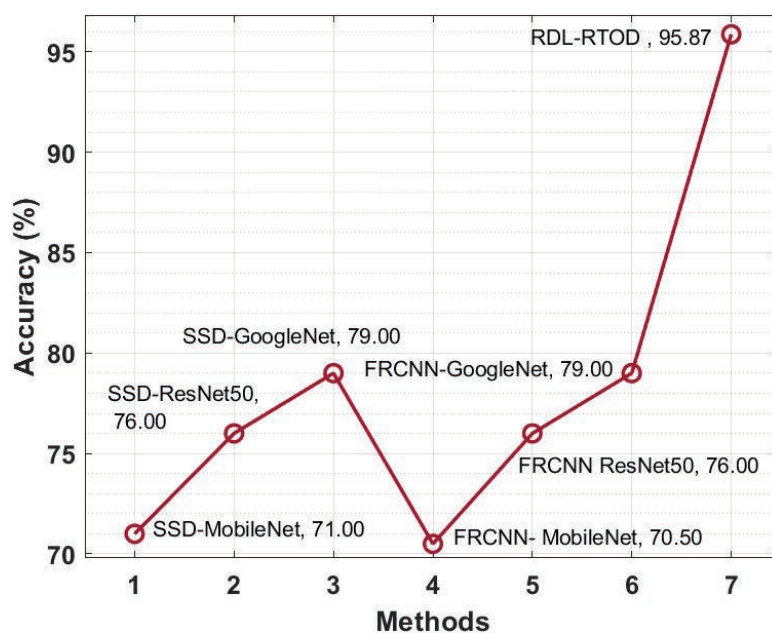


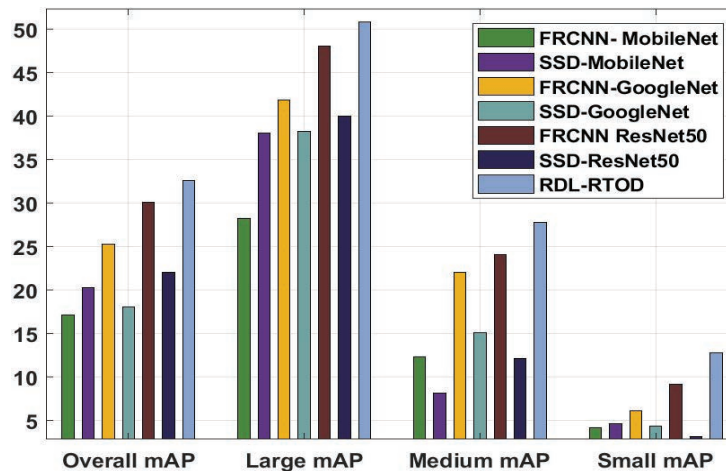
Figure 5 Accuracy analysis of RDL-RTOD model with existing approaches.

FRCNN-MobileNet and SSD-MobileNet techniques have shown ineffectual outcomes with the lower accuracy of 70.5% and 71%. At the same time, the SSD-ResNet-50 and FRCNN-ResNet50 models have obtained slightly enhanced accuracy of 76% and 76% respectively. Followed by, the SSD-GoogleNet and FRCNN-GoogleNet techniques have showcased somewhat reasonable accuracy of 79% and 79% respectively. However, the presented RDL-RTOD technique has resulted in maximum accuracy of 95.87%.

To assess the RDL-RTOD technique’s accuracy, Table 2 and Figure 5 present a quick comparison. As a result, 70.5 percent and 71% accuracy

Table 3 mAP analysis of RDL-RTOD technique with existing techniques

Methods	Overall mAP	Large mAP	Medium mAP	Small mAP
FRCNN- MobileNet	17.11	28.25	12.29	4.14
SSD-MobileNet	20.27	38.05	8.13	4.64
FRCNN-GoogleNet	25.25	41.88	22.09	6.13
SSD-GoogleNet	18.10	38.22	15.11	4.31
FRCNN ResNet50	30.07	48.03	24.09	9.13
SSD-ResNet50	22.09	40.05	12.12	3.14
RDL-RTOD	32.57	50.85	27.75	12.78

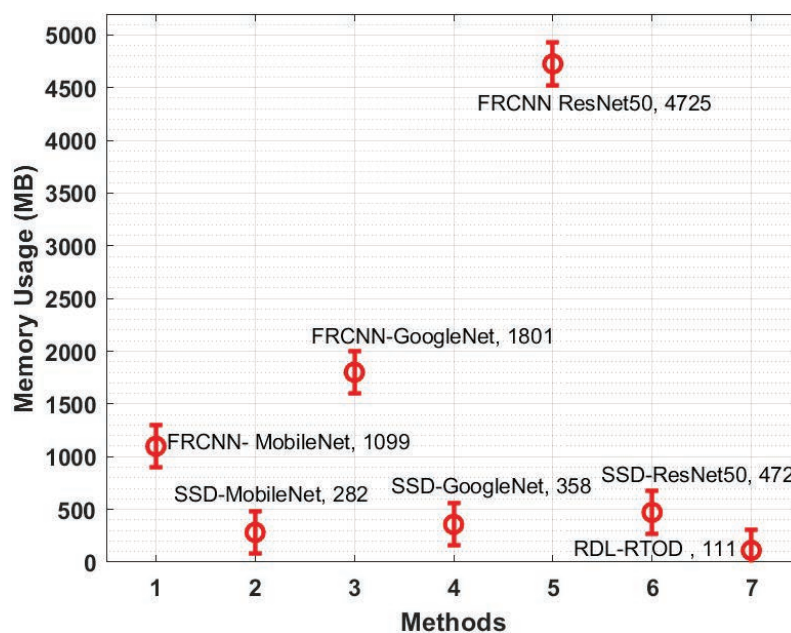
**Figure 6** mAP analysis of RDL-RTOD model.

for FRCNN-MobileNet and SSD-MobileNet were found. Simultaneously, the SSD-ResNet-50 and FRCNN-ResNet50 models achieved 76% and 76% accuracy. These approaches had a 79 percent accuracy for SSD and 79 percent accuracy for FRCNN-GoogleNet. However, the presented RDL-RTOD approach achieved 95.77% accuracy.

Table 3 and Figure 6 provide a detailed mean average precision (mAP) analysis of the RDL-RTOD technique with existing ones under varying resolutions. The results reported that the RDL-RTOD technique has accomplished proficient results with the least mAP of 17.11, 28.25, 12.29, and 4.14 under overall, large, medium, and small resolutions respectively. A comparison of the RDL-RTOD technique's mean average precision (mAP) with those already in use at various resolutions is shown in Table 3 and Figure 6. Overall, big, medium, and small resolutions all showed that the RDL-RTOD technique

Table 4 Memory usage analysis of RDL-RTOD model with state-of-art techniques

Methods	Memory Usage (MB)
FRCNN- MobileNet	1098.56
SSD-MobileNet	282.14
FRCNN-GoogleNet	1801.06
SSD-GoogleNet	358.08
FRCNN ResNet50	4725.00
SSD-ResNet50	472.00
RDL-RTOD	111.26

**Figure 7** Memory usage analysis of RDL-RTOD model with existing methods.

had excellent results with the lowest mAP of 17, 11, 28, 25, 12, 29, and 4, 14 points for mAP, respectively.

A detailed comparative study of the RDL-RTOD technique with other methods interms of memory usage is performed in Table 4 and Figure 7 [20]. The results portrayed that the FRCNN-ResNet50 and FRCNN-GoogleNet techniques have revealed incompetent performance with the higher memory usage of 4725MB and 1801.06MB. In addition, the SSD-ResNet-50 and FRCNN-MobileNet models have obtained somewhat reduced memory usage of 1098.56MB and 472MV respectively. Additionally, the SSD-GoogleNet

and SSD-MobileNet techniques have reported considerable memory usage of 358.08MB and 282.14MB respectively. However, the presented RDL-RTOD technique attained effectual outcome with the least memory usage of 111.26MB. Therefore, the proposed RDL-RTOD technique can be used for real time object detection process.

Table 4 and Figure 7 [20] present a detailed memory usage comparison of the RDL-RTOD technique with other methods. The results showed that the FRCNN-ResNet50 and FRCNN-GoogleNet techniques performed poorly, requiring 4725MB and 1801.06MB of memory, respectively. Furthermore, the SSD-ResNet-50 and FRCNN-MobileNet models used 1098.56MB and 472MV of memory, respectively. Furthermore, the SSD-GoogleNet and SSD-MobileNet techniques used a lot of memory, 358.08MB and 282.14MB, respectively. The presented RDL-RTOD technique, on the other hand, yielded a useful result while using the least amount of memory (111.26MB). As a result, the proposed RDL-RTOD technique can be applied to the process of real-time object detection. The major reason for the gaps among applications needs and technical abilities are because of the realtime key limitation such as object detection algorithm should be frequently hand tuned for certain types of context and object.it is complex to store and build a range of target object model, particularly if the object is different in its occurrence, and realtime object recognition demand higher computing power to identify individual object, much lesser while multiple target objects are included.

5 Conclusion

In this study, a new RDL-RTOD technique is developed for real time object detection and classification in UAV surveillance applications. The proposed RDL-RTOD technique encompasses a two-stage process namely object detection and objects classification. For detecting objects, YOLO-v2 with ResNet-152 technique is used and generates a bounding box for every object. In addition, the classification of detected objects takes place using OKELM model. Furthermore, the FFO algorithm is applied for tuning the weight parameter of the KELM model and thereby boosts the classification performance. A sequence of experimental validation processes takes place on the benchmark dataset and the results are inspected under several aspects. The experimental results highlighted the supremacy of the RDL-RTOD technique over the recent approaches interms of several performance measures. In future, advanced DL models with class attention mechanisms can be included to improve object detection performance.

References

- [1] Bhaskaranand, M, a. J. (n.d.). Low-complexity video encoding for UAV reconnaissance and surveillance. Proc. IEEE Military Communications Conference (MILCOM), pp. 1633-1638, 2011.
- [2] Lim, a. S. Monocular Localization of a moving person onboard a Quadrotor MAV. Proc. IEEE International Conference on Robotics and Automation (ICRA), pp. 2182–2189, 2015.
- [3] I. Sa, S. H. Outdoor flight testing of a pole inspection UAV incorporating highspeed vision. Springer Tracts in Advanced Robotics, pp. 107–121, 2015.
- [4] Christos Kyrkou, G. P. DroNet: Efficient convolutional neural network detector for realtime UAV applications. Design, Automation and Test in Europe Conference and Exhibition (DATE), 2018.
- [5] J. Gokul Anand, “Trust based optimal routing in MANET’s,” 2011 International Conference on Emerging Trends in Electrical and Computer Technology, Nagercoil, India, 2011, pp. 1150–1156, doi: 10.1109/ICET ECT.2011.5760293.
- [6] S.Divyabharathi, “Large scale optimization to minimize network traffic using MapReduce in big data applications”. International Conference on Computation of Power, Energy Information and Communication (ICCPEIC), pp. 193–199, April 2016. DOI: 10.1109/ICCPEIC.2016.7557196.
- [7] Paulraj, D 2020, ‘An Automated Exploring And Learning Model For Data Prediction Using Balanced CA-SVM’, Journal of Ambient Intelligence and Humanized Computing, Vol. 12, no. 5, April 2020, DOI: <https://doi.org/10.1007/s12652-020-01937-9>
- [8] Berlin, M.A., Tripathi, S. et al. IoT-based traffic prediction and traffic signal control system for smart city. Soft Computing (2021). <https://doi.org/10.1007/s00500-021-05896-x>
- [9] P. Subbulakshmi, “Mitigating eavesdropping by using fuzzy based MDPOP-Q learning approach and multilevel Stackelberg game theoretic approach in wireless CRN”, Cognitive Systems Research, Vol. 52, pp. 853–861, 2018. doi: 10.1016/j.cogsys.2018.09.021.
- [10] V. D. Ambeth Kumar, S. Malathi, Abhishek Kumar, Prakash M and Kalyana C. Veluvolu, “Active Volume Control in Smart Phones Based on User Activity and Ambient Noise,” Sensors 2020, 20(15), 4117. doi: 10.3390/s20154117.

- [11] Sambit Satpathy, Sanchali Das, Swapan Debbarma, “A new health-care diagnosis system using an IoT-based fuzzy classifier with FPGA”, *Journal of Supercomputing*, vol. 76, no. 8, pp. 5849–5861, 2020. doi: 10.1007/s11227-019-03013-2.
- [12] Jangwon, L., Wang, J., Crandall, D., Šabanovic, S. and Fox, G., 2017. Real-time object detection for unmanned aerial vehicles based on cloud-based convolutional neural networks. *Journal Concurrency and Computation: Practice and Experience*, 29(6).
- [13] F. S. Leira, T. A. Johansen, and T. I. Fossen, “Automatic detection, classification and tracking of objects in the ocean surface from UAVs using a thermal camera,” in *Proc. IEEE Aerospace Conference*, pp. 1–10, 2015.
- [14] J. Engel, J. Sturm, and D. Cremers, “Scale-aware navigation of a low-cost quadcopter with a monocular camera,” *Robotics and Autonomous Systems*, vol. 62, no. 11, pp. 1646–1656, 2014.
- [15] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, “Deep neural networks segment neuronal membranes in electron microscopy images,” in *Advances in neural information processing systems*, pp. 2843–2851, 2012.
- [16] Alam, M.S., Natesha, B.V., Ashwin, T.S. and Guddeti, R.M.R., 2019. UAV based cost-effective real-time abnormal event detection using edge computing. *Multimedia tools and Applications*, 78(24), pp. 35119–35134.
- [17] Vaddi, S., 2019. Efficient object detection model for real-time UAV applications (Doctoral dissertation, Iowa State University).
- [18] Zhang, H., Liptrott, M., Bessis, N. and Cheng, J., 2019, September. Real-time traffic analysis using deep learning techniques and uav based video. In *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)* (pp. 1–5). IEEE.
- [19] Nguyen, A.Q., Nguyen, H.T., Tran, V.C., Pham, H.X. and Pestana, J., 2021, January. A Visual Real-time Fire Detection using Single Shot MultiBox Detector for UAV-based Fire Surveillance. In *2020 IEEE Eighth International Conference on Communications and Electronics (ICCE)* (pp. 338–343). IEEE.
- [20] Lai, Y.C. and Huang, Z.Y., 2020. Detection of a Moving UAV Based on Deep Learning-Based Distance Estimation. *Remote Sensing*, 12(18), p. 3035.
- [21] Dong, J., Ota, K. and Dong, M., 2021. UAV-based Real-Time Survivor Detection System in Post-disaster Search and Rescue Operations. *IEEE Journal on Miniaturization for Air and Space Systems*.

- [22] Redmon, J., and Farhadi, A. (2017). YOLO9000: Better, Faster, Stronger. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 6517–6525.
- [23] Loey, M., Manogaran, G., Taha, M.H.N. and Khalifa, N.E.M., 2021. Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-152 for medical face mask detection. *Sustainable cities and society*, 65, p. 102600.
- [24] Lu, J., Huang, J. and Lu, F., 2019. Distributed kernel extreme learning machines for aircraft engine failure diagnostics. *Applied Sciences*, 9(8), p. 1707.
- [25] Ding, G., Dong, F. and Zou, H., 2019. Fruit fly optimization algorithm based on a hybrid adaptive-cooperative learning and its application in multilevel image thresholding. *Applied Soft Computing*, 84, p. 105704.
- [26] <http://aiskyeye.com/download/>
- [27] Sun, C., Zhan, W., She, J. and Zhang, Y., 2020. Object detection from the video taken by drone via convolutional neural networks. *Mathematical Problems in Engineering*, 2020.

Biographies



C. Prasanna Ranjith Obtained Ph.D. in Computer Science from Bharathidasan University, Trichy, India in Dec 2017. Received Bachelor, Master of Science and Master of Philosophy in computer science in 1995,1997 & 2004 from Bharathidasan University, Trichy, India. Have more than 23 years of teaching experience at different Colleges and Universities in India, Libya, and Sultanate of Oman. Presently a Faculty & Research Chair, department of Information Technology at University of Technology and Applied Sciences, Shinas, Oman since October 2011. Organized various Workshops, Seminars and symposiums towards professional Research and development. Published many research papers in international journals and conferences of

high repute. Hold ample experience in the field of Web Designing having widespread familiarity in ASP.Net (C#) & MS SQL Server. Awarded BEST FACULTY for 3 consecutive years. My research interests include Machine Learning, Deep Learning, Nature Inspired Algorithms, Soft Computing, Parallel Algorithms, Genetic Algorithms and Ad Hoc Networks.



Bhalchandra M. Hardas has done his M.Tech and Ph D in Electronics Engineering from Rashtsant Tukadoji Maharaj Nagpur University, Nagpur. His research contribution includes development of statistical maximum value distribution algorithm for AWGN, Rayleigh and Rician channel in MIMO-OFDM. His current interests are in FPGA implementation of novel PAPR reduction algorithm. He is member of ISTE and IETE. He is recipient of Best teacher award of RTM Nagpur University in year 2017–2018. Presently he is working as Assistant Professor in Electronics Engineering department. He is coordinator of IPR cell of Shri Ramdeobaba college of Engineering and Management, Nagpur.



M. Syed Khaja Mohideen Completed Undergraduate from MS University in 1997, Masters degree in Computer Science in 2000, MPhil(Computer Science) in 2004 and the PhD degree in Computer Science in 2019 from Bharatidasan University Trichy. Worked as Assistant Professor in Computer Science at Bishop Heber College till 2007 and Omar Mukthar University, Libya 2007 to 2011. Presently working in University of Technology and Applied Sciences – Salalah, Sultanate of Oman. Current research interest includes ad-hoc networks.



N. Nijil Raj, currently working as professor and head, department of CSE, Younus College of engineering and technology, Kollam, affiliated to APJ technological university, Thiruvananthapuram, Kerala, He has published more than 15 papers in national and international journals. He was post graduated from M. S. University, Tamilnadu in M.Tech, MCA and MBA from MG University Kottayam, Kerala. His area interest is bioinformatics, Machine Learning and AI, and Image Processing. He is having 18 years of teaching experience in UG and PG level. He has completed the Doctoral Degree from M.S. University, Tamilnadu, in Computer Science and Information Technology.



Nismon Rio Robert is currently working as an Assistant Professor of Computer Science, CHRIST (Deemed to be University), Bangalore, India, since 2019. He holds a Ph.D. Degree in Computer Science from Bishop Heber College affiliated to Bharathidasan University, Tiruchirappalli, India in 2019. He received financial support from University Grants Commission (UGC) of the research work. His research interest is route optimization in wireless networks, Mobile Ad-hoc Networks, Internet of Things and other related topics. He has presented 30 papers in National and International conferences/journals. Moreover, he has published 6 Indian Patents in the field of communication technologies. He has delivered technical lectures in various institutions.



Prakash Mohan, SM IEEE, Professor is working in the Department of Computer Science and Engineering Karpagam College of Engineering, Coimbatore, Tamil Nadu, India. He received his Doctor of Philosophy in 2014 from Jawaharlal Nehru Technological University Hyderabad. He has more than 20+ years of experience in teaching and research. His area of interest includes Data Analytics, Big Data, and Machine Learning. He has published more than 50 research papers in International Journals/ Conferences. He has served as a lead guest editor for the journals Inderscience, EAI Endorsed Transition, Bentham publisher, and Tech science press. He is an Editorial

Review Board of IGI Global. He has reviewed journals from the publishers IEEE, Elsevier, Springer, Taylor & Francis, and Inderscience. He has received awards from the Computer Society of India towards the Highest hosting CSI Events and is an Active CSI member in Coimbatore Chapter. He received grants from CSIR and CSI to conduct Conference, Seminar and Students State Level Convention. He is a member of IEEE, Association for Computing Machinery, Computer Society of India, ISTE, and IAENG.