

Modelling of Intelligent Object Detection and Classification using Aquila Optimizer with Deep Learning on Surveillance Videos

V. Saikrishnan^{1,*}, Dr. M. Karthikeyan²

¹Department of Computer and Information Science,
Faculty of Science, Annamalai University,
Tamil Nadu, India.

saikrishnan1074@gmail.com

²Department of Computer and Information Science,
Faculty of Science, Annamalai University,
Tamil Nadu, India.

karthiaucse@gmail.com

Abstract—Object Detection (OD) in surveillance video is the way of automatically detecting and tracking object classes of interest within the video recording. It includes the application of a Computer Vision (CV) technique to analyze the video frame and identify the classes of objects or the presence of specific objects. Various OD techniques are used to find objects within the footage video. This algorithm analyzes the visual feature of the frames and employs Machine Learning (ML) approaches namely Deep Neural Network (DNN), to detect and track objects. It is worth mentioning that the accuracy and performance of OD in surveillance video depends on factors including the choice of algorithms and models, the availability of labelled training data, and the quality of the video frame for the specific object of interest. This study introduces a new modeling of Intelligent Object Recognition and Classification by employing Aquila Optimizer with Deep Learning (IODC-AODL) approach in Surveillance Video. The goal of the IODC-AODL technique is to integrate the DL model with the hyperparameter tuning process for object detection and classification. In the proposed IODC-AODL approach, a Faster RCNN method is enforced for the process of OD. Next, Long Short-Term Memory (LSTM) networking approach is implemented for the object classification process. At last, the AO approach is enforced for the optimum hyperparameter tuning of the LSTM network and it assists in improving the classifier rate. A widespread simulation sets are performed to exhibit the superior performance of the IODC-AODL approach. The experimental result analysis portrayed the supremacy of the IODC-AODL algorithm over other models.

Keywords- Computer vision; Object detection; Aquila optimizer; Deep learning; Object classification.

I. INTRODUCTION

Artificial Intelligence (AI) technique has covered the way for computers to think like humans. ML creates a way more balanced by integrating with the learning and training components [1]. The accessibility of large datasets and higher performance of computers leads the light to DL concepts that are automatically extracted features or the factors of variation, which differentiate the objects from each other [2]. In the current world, video surveillance data have much social significance among the different data sources which provide terabytes of big data [3]. The broad accessibility of surveillance data from the cameras deployed in industrial plants, academic institutions, commercial firms, and residential regions gives towards private data whereas cameras located in public places like public transportation, spiritual places, and city centres contribute to public data [4]. Surveillance video analysis includes action detection, OD, and classification of detected actions into categories such as normal or anomalous [5].

The problem description of OD is used to define whether the objects are positioned in a given image (object localization) and which class for every object refers to object classification [6]. Thus, the pipeline of the conventional OD method could be generally separated into three phases namely, classification, feature extraction, and informative region selection [7]. The OD method is used to determine the class instance to which the object belongs and the object's position is estimated via outputting the bounding box around the objects. The process of single-class OD is used for detecting a single instance of the class from the image, while multi-class OD is the process of detecting the class of each object existing in the image [8]. Various challenges like scale, changing illumination conditions, poses, and full or partial occlusion are required to be managed when performing the OD. Deep Convolutional Neural Networks (DCNNs) approach is broadly utilized for the detection of objects. The CNN approach is a category of Feedforward Neural Networks (FFNN) and works under the principle of weight sharing [9]. Furthermore, a classifier is required to differentiate

a target object from entirely other classes and to create representations, more informative, hierarchical, and semantic for visual recognition [10].

This study introduces a new modeling of Intelligent Object Recognition and Classification by employing Aquila Optimizer with Deep Learning (IODC-AODL) technique in Surveillance Video. The goal of the IODC-AODL technique is to integrate the DL model with the hyperparameter tuning process for object detection and classification. In the proposed IODC-AODL approach, a Faster RCNN method is enforced for the process of OD. Next, Long Short-Term Memory (LSTM) networking approach is implemented for the object classification process. At last, the AO approach is enforced for the optimum hyperparameter tuning of the LSTM network and it assists in improving the classifier rate. A widespread simulation sets are performed to exhibit the superior performance of the IODC-AODL approach.

II. RELATED WORKS

Ezuma et al. [11] developed an approach that used energy transient signals. This algorithm can be more effective with noise and may deal with modulation techniques. In the initial stage, the normalized energy trajectory is produced from the energy-time-frequency dispersal of the raw control signals. A significant feature was selected by carrying out Neighbourhood Component Analysis (NCA) for keeping the computation price of the model lower. In the next stage, the features selected are fed to ML techniques for classification. Once the attacker does not have access to a model gradient or structure by employing the patch-based blackbox adversarial attacks with Reinforcement Learning (RL), the authors in [12], leveraged the vulnerability and the DL method's decision-making ability in a gait-based independent surveillance system. Fredianelli et al. [13] devised an effective tool based on lower cost camera and vehicle detection and counting method by employing ML algorithms, by the requirement of noise assessment mechanism.

The authors in [14], used diverse versions of the YOLOv6 architectures (for example, YOLOv6n, YOLOv6t, YOLOv6m, YOLOv6s, YOLOv6l relu, and YOLOv6l). The objective is to devise a dependable and effective tool for PB recognition. Chandrakar et al. [15] introduce state-of-the-art methods for automated Object Tracking (OT) and Moving OD (MOD) using RBF-FDLNN and CFR techniques. Initially, the noise that exists in video frames is filtered using a GESFA filter. Consequently, the features required in the frame were extracted and fed into the classifiers to perform the OD task. Mendoza et al. [16] designed an insect detection scheme comprising a manual-focusing camera, a less-power, lower-price singular-board computer, and a trained DL algorithm. The technique was authorized using a live visual feed. Vennila and Balamurugan [17] address the abovementioned problems by presenting a rough set architecture

that recognizes the human object by implementing the bounding box generation method and the rough set classifiers.

III. THE PROPOSED MODEL

In the present study, the design and development of the IODC-AODL technique on surveillance videos is offered. The goal of the IODC-AODL technique is to integrate the DL model with the hyperparameter tuning process for object detection and classification. It includes three stages of operations namely Faster RCNN-based OD, LSTM-based classification, and AO-based hyperparameter tuning. Fig. 1 shows the work flow of the IODC-AODL methodology.

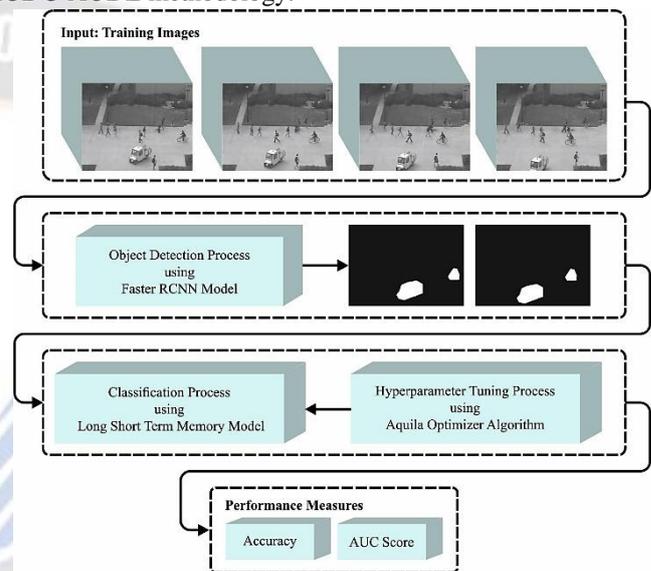


Figure 1. Workflow of IODC-AODL method

A. Object Detection Using Faster RCNN Model

The fast RCNN model is used to effectually detect the objects. Usually, the Fast RCNN works under the principles of three major parts namely generation of bounding boxes, classifier classification and object position correction through regressor, and feature data extraction from input images [18]. The input image was employed by the CNN for the extraction of features, and the convolution features attained are exploited as an input of the Region Proposal Network (RPN) that sequentially generates the region proposal. The classification layer's main role is to define the objects in the bounding boxes as either background or object, whereas the regression layer's main role is to forecast the parameter of region proposals respective to the anchor point of the bounding boxes. The region formed by RPN was mapped to the convolution feature map based on the position to generate a Region of Interest (RoI).

Next, the RoI pooling function is implemented to divide the mapped RoI region into similar size blocks, and later the max-pooling function is implemented for adjusting the bounding box size in all the regions. Then, the data of the bounding box in all the regions are transferred to the next layer of the networks, viz.,

the FC layer, later the location of the corrected bounding box and the label classification score are outputted by the softmax function.

(1) loss of bounding box location correction at recognition and classification; and (2) loss of classification and bounding box regression of the regional networks are two major elements of the loss function of Fast RCNN and these are determined as follows:

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

In Eq. (1), p_i shows the predictive probability of the anchor to the object, and i denotes the index of the anchor number in all the smaller batches of data. p_i^* indicates the label value of the category that is 0 or 1, 0 implies a false object, and 1 implies a true object. λ refers to the weight coefficient, N_{cls} denotes the classification loss parameter, N_{reg} indicates the regression loss parameter, L_{cls} denotes the classification loss in OD, and $L_{cls}(p_i, p_i^*)$ shows the loss of logarithmic between the non-object and detection object that can be evaluated using Eq. (2). R denotes the function called smooth (x), and the expression is given as follows:

$$L_{cls}(p_i, p_i^*) = -\log [p_i^* p_i + (1 - p_i^*)(1 - p_i)] \quad (2)$$

$$smooth_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \quad (3)$$

Where $L_{reg}(t_i, t_i^*)$ denotes the regression loss inside the OD that can be denoted as $L_{reg}(t_i, t_i^*) = R(t_i - t_i^*)$, t_i indicates the predicted coordinate, t_i^* shows the coordinate of the detection object, and $x = t_i - t_i^*$.

B. LSTM-based Classification

For classifying the detected objects, the LSTM method is enforced. LSTM is a kind of RNN. The RNN could not learn relevant data regarding the input dataset once the input gap is larger and could deal with the sequences of the long input [19], however, LSTM could handle long-term dependency by presenting gate functions in the cell structures. Thus, the LSTM method was introduced to attain the feature in the temporal dimension.

Fig. 2 represents the framework of LSTM. In contrast to prior RNN, LSTM adds the concept of cell states, involving the input, forget, and output gates. Then, element-wise multiplication was used between them for controlling what amount of historical memory can be retained. Here, regular crop changes are utilized as a time dimension, and twelve months of the dataset are inputted into the LSTM for extracting the effects of the crop growth process on the yields. Especially, the forget gate can be tested as given in Eq. (4), commonly used to decide the forgetting or retention of data.

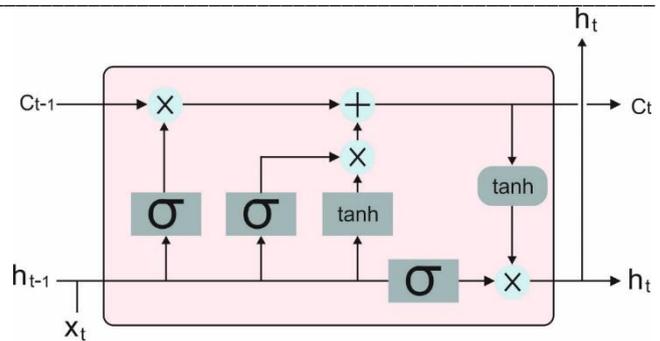


Figure 2. Architecture of LSTM

The x_t input data of the existing layer and h_{t-1} hidden data of the prior layer were transferred into the sigmoid activation function for simultaneous processing. These results would be in the range [0, 1]. The nearer it is to 1, the more it would be retained.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (4)$$

Additionally, the formula of the input gate is given in Eqs. (5) & (6), extensively applied for updating the data of the existing layer. At the same time, the hidden data h_{t-1} of the prior layer and the input data x_t of the existing layer were directed into the sigmoid function. The processing results would be in the [0, 1] range. The nearer it is to 1, the most crucial it would be. Lastly, the sigmoid activation function's output values are multiplied by the Tanh function's output values which determines data in the output values of Tanh is crucial and must be maintained.

$$i_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_i) \quad (5)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (6)$$

The output gate, given in Eqs. (7) to (9) define the values of the next hidden state that has prior inputted data. The data x_t from the existing layer input and the hidden data h_{t-1} from the prior layer are transmitted to the sigmoid activation function. Next, the new cell state is transferred to the Tanh activation function. Later, the hidden layer is exploited as an existing cell output, and the new hidden and cell states are transmitted to the next time step.

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (7)$$

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (8)$$

$$h_t = O_t * \tanh(C_t) \quad (9)$$

C. AO-based Hyperparameter Tuning

In this work, the AO algorithm selects the hyperparameter value of the LSTM. The AO is a SI technique where Aquila has four different hunting strategies for the prey [20]. Similar to other MH approaches, AO implements the optimization technique by employing ERP and ETP and will be lastly converging towards the last optimum solution. The AO technique summary is given below.

Step1: Exploring the search space widely. The mathematical modelling of these behaviours is given in the following:

$$X(t+1) = X_{best}(t) \times \left(1 - \frac{t}{T}\right) + (X_m(t) - X_{best}(t) \times rand) \quad (10)$$

$$X_m(t) = \frac{1}{N} \sum_{k=1}^N X_i(t) \quad (11)$$

From the expression, $rand$ is a randomly generated value within $[0,1]$, $X_m(t)$ refers to the average position of each agent in the existing iteration, $X_{best}(t)$ shows the fittest location, N indicates the population size and t and T show the existing iteration and the maximal amount of iterations, correspondingly.

Step2: Narrow exploration. The updating position can be expressed in the following expression:

$$X(t+1) = X_{best}(t) \times LP(D) + X_R(t) + (y - x) \times rand \quad (12)$$

In Eq. (12), $X(t)$ shows the random location of Aquila, D stands for the dimension size, and $LP(D)$ indicates the Levy flight function that can be formulated below:

$$LP(D) = s \times \frac{u \times \sigma}{|v|^\beta} \quad (13)$$

The σ parameter can be attained based on the subsequent formula:

$$\sigma = \left[\frac{r(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{r\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \right] \quad (14)$$

In Eq. (14), s , u , and v denotes the random number within $[0, 1]$, β are fixed value equivalent to 0.01 and 1.5, correspondingly, y and x indicates the spiral shape to explore the search space that can be described as follows:

$$\begin{cases} x = r \times \sin(\theta) \\ y = r \times \cos(\theta) \\ r = r_1 + 0.00565 \times D_1 \\ \theta = -w \times D_1 + \frac{3 \times \pi}{2} \end{cases} \quad (15)$$

In Eq. (15), r_1 implies the number of search cycles in the range of 1 and 20, and D_1 indicates the integer from 1 to D and is equivalent to 0.005.

Step3: Extensive exploitation: Aquila vertical landing towards attacking the prey. These behaviours are described in the following:

$$X(t+1) = (X_{best}(t) - X_m(t)) \times \alpha - rand + ((UB - LB) \times rand + LB) \times \delta \quad (16)$$

In Eq. (16), α and δ denote the ETP adjustment parameter which is equivalent to 0.1, and UB and LB correspondingly indicate the upper and lower boundaries of the problems.

Stage 4: Limited exploitation: The mathematical modelling of moving and catching prey is given below:

$$\begin{cases} X(t+1) = QP \times X_{best}(t) - (G_1 \times X(t)) \times rand - G_2 \times LP(D) + rand \times G_1 \\ QF(t) = t^{\frac{2 \times rand - 1}{(1-T)^2}} \\ G_1 = 2 \times rand - 1 \\ G_2 = 2 \times \left(1 - \frac{t}{T}\right) \end{cases} \quad (17)$$

In Eq. (17), $X(t)$ denotes the existing position and $QF(t)$ shows the quality function value for balancing the searching process, parameter G_1 represent the movement of Aquila once tracing the prey is randomly generated value within $[-1,1]$. G_2 denotes the flight slope while pursuing prey that linearly drops from 2 to 0. AO has high power in the ERp however it is a bit slower in the ETp and for such reasons, it could not accurately search for the optimum solution with higher power.

The AO method derives a fitness function for attaining high efficiency of classification. It defines a positive integer to indicate the solution candidate's enhanced results. The decrease in classifier error rate is assumed as an FF:

$$fitness(x_i) = \frac{ClassifierErrorRate(x_i) \times No. of misclassified samples}{Total No. of samples} * 100 \quad (18)$$

IV. PERFORMANCE VALIDATION

This section examines the accomplishment of the IODC-AODL method on the UCSDPed2 dataset. The dataset comprises two sub datasets with 360 frames as portrayed in Table 1. Fig. 3 represents the sample original and ground truth images.

TABLE 1. DETAILS OF THE DATASET

Dataset	Test Ped	Frames Number	Time (sec)
UCSDped2	Pedestrian1 Dataset	360	12
	Pedestrian2 Dataset		

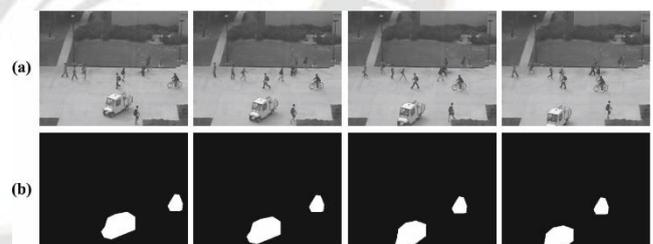


Figure 3. a) Original Image b) Ground Truth Images

The average results analysis of the IODC-AODL approach on two datasets is given in Table 2 and Fig. 4 [22-25]. The outputs show that the IODC-AODL approach gains increasing accuracy values over other models. For instance, on the SPed-1 dataset, the IODC-AODL model offers increasing accuracy of 99.09% whereas the CIHSART-ODT, DLADT, Region CNN, and FR-CNN models obtain decreasing accuracy of 98%, 97%, 97%, and 85% respectively. Additionally, on the SPed-2 dataset, the IODC-AODL technique offers a raising accuracy of 97.13% while the CIHSART-ODT, DLADT, Region CNN, and FR-CNN techniques attain decreasing accuracy of 91%, 90%, 87%, and 82% respectively.

TABLE 2. AVERAGE ACCURACY OUTCOME OF IODC-AODL APPROACH WITH OTHER METHODS ON TWO DATASETS

Methods	IODC-AODL	CIHSAR T-ODT	DLADT	Region CNN	FR-CNN
Surveillance Ped1	99.09	98.00	97.00	97.00	85.00
Surveillance Ped2	97.13	91.00	90.00	87.00	82.00

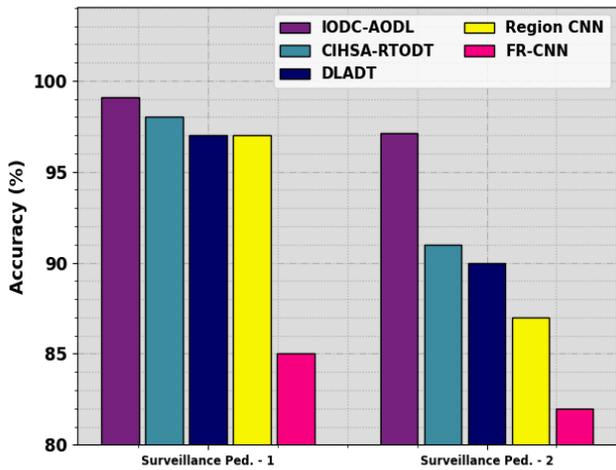


Figure 4. Average accuracy outcome of the IODC-AODL method on two datasets

The AUC results analysis of the IODC-AODL model on two datasets is given in Table 3 and Fig. 5. The outputs portrayed that the IODC-AODL model gains to increase AUC values over other techniques. For instance, on SPed-1 dataset, the IODC-AODL approach offers to increase the AUC of 99.35% whereas the MP-PCA, SF, SFMP-PCA, M-DT, A-MDN, AD-VAE, and CIHSART-ODT approach obtain reducing AUC of 61.01%, 66.74%, 67.25%, 82.05%, 91.71%, 95.39%, and 97.12% respectively. Moreover, on SPed-2 datasets, the IODC-AODL system offers raising AUC of 97.17% whereas the MP-PCA, SF, SFMP-PCA, M-DT, A-MDN, AD-VAE, and CIHSART-ODT system obtain diminishing AUC of 69.92%, 55.96%, 61.33%, 82.99%, 91.25%, 92.47%, and 93.92% correspondingly.

TABLE 3. AUC OUTPUT OF IODC-AODL MODEL WITH OTHER TECHNIQUES ON TWO DATASETS

Models	SPed-1	SPed-2
MP-PCA	61.01	69.92
SF	66.74	55.96
SFMP-PCA	67.25	61.33
M-DT	82.05	82.99
A-MDN	91.71	91.25
AD-VAE	95.39	92.47
CIHSART-ODT	97.12	93.92
IODC-AODL	99.35	97.17

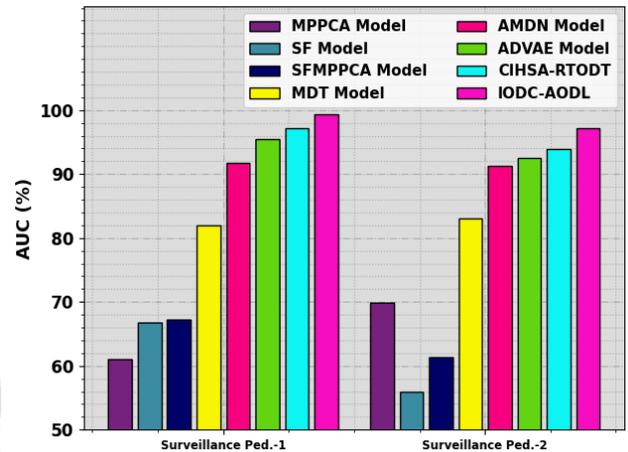


Figure 5. AUC output of IODC-AODL model with other techniques on two datasets

The comparative Running Time (RT) results of the IODC-AODL methodology are shown in Table 4 and Fig. 6.

TABLE 4. RT OUTPUT OF IODC-AODL MODEL WITH OTHER TECHNIQUES ON TWO DATASETS

Models	Pedestrian1	Pedestrian2
M-DT	20.67	22.91
SCLF	20.05	18.47
A-MDN	11.97	13.01
AD-VAE	04.04	05.97
CIHSART-ODT	02.60	04.10
IODC-AODL	02.08	02.23

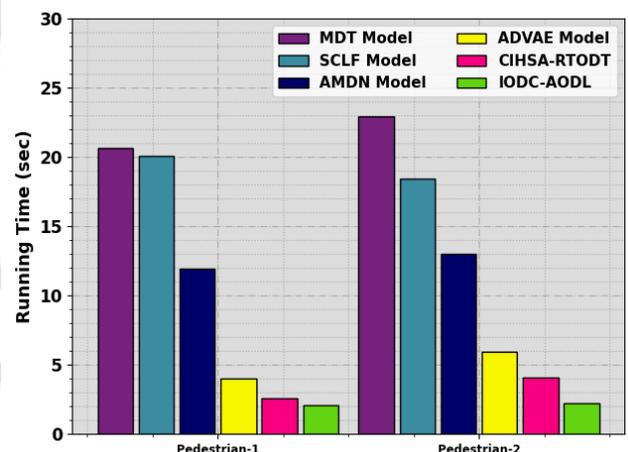


Figure 6. RT output of IODC-AODL model on two datasets

The outputs illustrate the improved performance of the IODC-AODL methodology with the least RT values. For instance, on the SPed-1 dataset, the IODC-AODL model accomplishes reduced 2.08s whereas the CIHSART-ODT, DLADT, Region CNN, and FR-CNN models attain increased RT of 20.67s, 20.05s, 11.97s, 4.04s, and 2.60s correspondingly. Furthermore, on the SPed-2 dataset, the IODC-AODL technique

accomplishes decreased 2.08s whereas the CIHSART-ODT, DLADT, Region CNN, and FR-CNN techniques achieved to increase RT of 22.91s, 18.47s, 13.01s, 5.97s, and 4.10s correspondingly.

The ROC investigation of the IODC-AODL approach on the SPed-1 dataset is related with existing techniques in Table 5 and Fig. 7. Besides, the SF, A-MDN, and AD-VAE approaches have obtained the least object detection and classification results. Also, the CIHSART-ODT system has managed to obtain manageable performance. The outcome exhibits the maximum performance of the IODC-AODL system over other methods.

TABLE 5. ROC ANALYSIS OF IODC-AODL MODEL WITH RECENT TECHNIQUES ON SPED-1

ROC	SF	A-MDN	AD-VAE	CIHSART-ODT	IODC-AODL
10	18.08	23.14	22.05	47.28	49.73
20	32.26	46.20	43.07	70.34	74.50
30	43.57	64.02	69.14	90.02	98.01
40	54.22	76.05	78.99	92.93	94.98
50	61.74	83.94	92.39	94.67	99.35
60	69.23	91.03	94.79	99.65	99.94
70	86.50	99.29	100.60	99.49	99.84
80	88.61	92.89	98.52	97.57	99.78
90	88.46	98.47	99.56	101.48	99.95
100	90.16	94.49	97.06	97.80	99.53

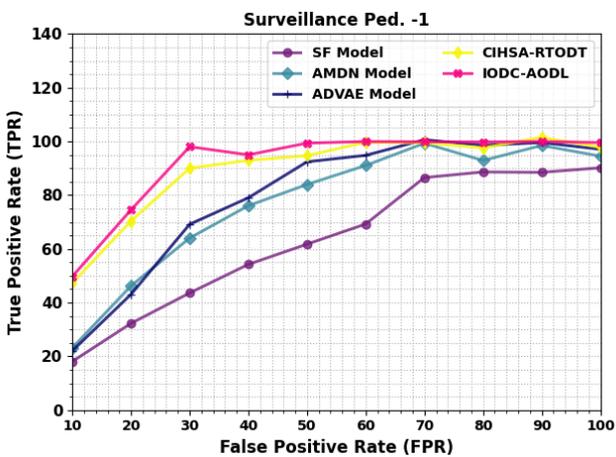


Figure 7. Fig. 7. ROC analysis of IODC-AODL model on SPed-1

The ROC study of the IODC-AODL method on the SPed-2 dataset is related with current techniques in Table 6 and Fig. 8. Further, the SF, A-MDN, and AD-VAE methods have achieved least results. Also, the CIHSART-ODT approach acquired manageable performance. The outputs showed enhanced achievement of the IODC-AODL approach over other models.

TABLE 6. ROC ANALYSIS OF IODC-AODL METHOD WITH RECENT APPROACHES ON SPED-2

ROC	SF	A-MDN	AD-VAE	CIHSART-ODT	IODC-AODL
10	20.90	27.12	18.13	27.08	57.04
20	27.63	47.45	27.65	63.14	64.50
30	41.68	57.13	70.98	79.81	83.14
40	56.95	73.31	83.38	92.88	97.90
50	74.86	89.38	87.06	98.52	99.39
60	87.13	93.34	95.26	96.22	99.85
70	98.16	97.67	99.08	97.13	99.87
80	98.38	99.21	99.50	98.25	99.98
90	98.44	99.50	97.98	98.60	99.84
100	98.75	98.80	98.94	99.90	99.94

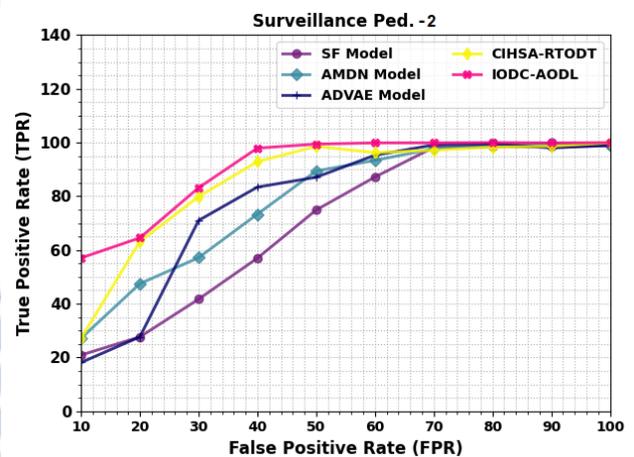


Figure 8. ROC analysis of IODC-AODL method on SPed-2

Thus, the IODC-AODL method can be used for robust object detection and classification on surveillance videos.

V. CONCLUSION

In this study, the IODC-AODL technique is designed and developed for video surveillance system. The goal of the IODC-AODL technique is to integrate the DL model with the hyperparameter tuning process for object detection and classification. In the proposed IODC-AODL model, a Faster RCNN model is employed for the OD technique. Next, in the subsequent stage, the LSTM network is exploited for the object classification process. Lastly, the AO approach is applied for the optimum hyperparameter tuning of the LSTM network and it assists in enhancing the classifier rate. An extensive set of simulations are performed to exhibit superior performance of the IODC-AODL approach. The simulation outputs portrayed the enhancement of the IODC-AODL method over other models.

REFERENCES

- [1] Taha, B. and Shoufan, A., 2019. Machine learning-based drone detection and classification: State-of-the-art in research. *IEEE access*, 7, pp.138669-138682.

- [2] Teja, Y.D., 2023. Static object detection for video surveillance. *Multimedia Tools and Applications*, pp.1-13.
- [3] Elhoseny, M., 2020. Multi-object detection and tracking (MODT) machine learning model for real-time video surveillance systems. *Circuits, Systems, and Signal Processing*, 39, pp.611-630.
- [4] Ahn, H. and Cho, H.J., 2022. Research of multi-object detection and tracking using machine learning based on knowledge for video surveillance system. *Personal and Ubiquitous Computing*, pp.1-10.
- [5] Prati, A., Shan, C. and Wang, K.I.K., 2019. Sensors, vision and networks: From video surveillance to activity recognition and health monitoring. *Journal of Ambient Intelligence and Smart Environments*, 11(1), pp.5-22.
- [6] Dabhade, V., Dhawalshankh, D., Thakare, A., Kulkarni, M. and Ambekar, P., 2023. A Smart System for Obstacle Detection to Assist Visually Impaired in Navigating Autonomously Using Machine Learning Approach. *Artificial Intelligence Applications and Reconfigurable Architectures*, pp.137-149.
- [7] Sun, W., Dai, L., Zhang, X., Chang, P. and He, X., 2021. RSOD: Real-time small object detection algorithm in UAV-based traffic monitoring. *Applied Intelligence*, pp.1-16.
- [8] Kinaneva, D., Hristov, G., Raychev, J. and Zahariev, P., 2019, May. Early forest fire detection using drones and artificial intelligence. In *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 1060-1065). IEEE.
- [9] Kim, H., 2022. Multiple vehicle tracking and classification system with a convolutional neural network. *Journal of Ambient Intelligence and Humanized Computing*, 13(3), pp.1603-1614.
- [10] Sampedro, C., Rodriguez-Ramos, A., Bavle, H., Carrio, A., de la Puente, P. and Campoy, P., 2019. A fully-autonomous aerial robot for search and rescue applications in indoor environments using learning-based techniques. *Journal of Intelligent & Robotic Systems*, 95, pp.601-627.
- [11] Ezuma, M., Erden, F., Anjinappa, C.K., Ozdemir, O. and Guvenc, I., 2019, March. Micro-UAV detection and classification from RF fingerprints using machine learning techniques. In *2019 IEEE Aerospace Conference* (pp. 1-13). IEEE.
- [12] Maqsood, M., Yasmin, S., Gillani, S., Aadil, F., Mehmood, I., Rho, S. and Yeo, S.S., 2023. An autonomous decision-making framework for gait recognition systems against adversarial attack using reinforcement learning. *ISA transactions*, 132, pp.80-93.
- [13] Fredianelli, L., Carpita, S., Bernardini, M., Del Pizzo, L.G., Brocchi, F., Bianco, F. and Licitra, G., 2022. Traffic flow detection using camera images and machine learning methods in ITS for noise map and action plan optimization. *Sensors*, 22(5), p.1929.
- [14] Bist, R.B., Subedi, S., Yang, X. and Chai, L., 2023. A novel YOLOv6 object detector for monitoring piling behavior of cage-free laying hens. *AgriEngineering*, 5(2), pp.905-923.
- [15] Chandrakar, R., Raja, R., Miri, R., Sinha, U., Kushwaha, A.K.S. and Raja, H., 2022. Enhanced the moving object detection and object tracking for traffic surveillance using RBF-FDLNN and CBF algorithm. *Expert Systems with Applications*, 191, p.116306.
- [16] Mendoza, Q.A., Pordesimo, L., Neilsen, M., Armstrong, P., Campbell, J. and Mendoza, P.T., 2023. Application of Machine Learning for Insect Monitoring in Grain Facilities. *AI*, 4(1), pp.348-360.
- [17] Vennila, T.J. and Balamurugan, V., 2023. A Rough Set Framework for Multi-Human Tracking in Surveillance Video. *IEEE Sensors Journal*.
- [18] Wang, H. and Xiao, N., 2023. Underwater Object Detection Method Based on Improved Faster RCNN. *Applied Sciences*, 13(4), p.2746.
- [19] Liu, F., Jiang, X. and Wu, Z., 2023. Attention Mechanism-Combined LSTM for Grain Yield Prediction in China Using Multi-Source Satellite Imagery. *Sustainability*, 15(12), p.9210.
- [20] Liu, Q., Kosarirad, H., Meisami, S., Alnowibet, K.A. and Hoshyar, A.N., 2023. An Optimal Scheduling Method in IoT-Fog-Cloud Network Using Combination of Aquila Optimizer and African Vultures Optimization. *Processes*, 11(4), p.1162.
- [21] <http://www.svcl.ucsd.edu/projects/anomaly/dataset.htm>
- [22] Pustokhina, I.V., Pustokhin, D.A., Vaiyapuri, T., Gupta, D., Kumar, S. and Shankar, K., 2021. An automated deep learning based anomaly detection in pedestrian walkways for vulnerable road users safety. *Safety Science*, 142, p.105356.
- [23] Xu, M., Yu, X., Chen, D., Wu, C. and Jiang, Y., 2019. An efficient anomaly detection system for crowded scenes using variational autoencoders. *Applied Sciences*, 9(16), p.3337.
- [24] Murugan, B.S., Elhoseny, M., Shankar, K. and Uthayakumar, J., 2019. Region-based scalable smart system for anomaly detection in pedestrian walkways. *Computers & Electrical Engineering*, 75, pp.146-160.
- [25] Alotaibi, M.F.; Omri, M.; Abdel-Khalek, S.; Khalil, E.; Mansour, R.F. Computational Intelligence-Based Harmony Search Algorithm for Real-Time Object Detection and Tracking in Video Surveillance Systems. *Mathematics* 2022, 10, 733. <https://doi.org/10.3390/math10050733>