INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Check for updates

OPEN ACCESS

Received: 26-05-2023

Accepted: 10-06-2023

Published: 27-06-2023

Citation: Saravanakumar S, Lingaraj M (2023) An Enriched Model of Neutrosophical Fuzzy and Grasshopper Convolutional Neural Network Based Moving Object Detection and Classification to Improve Video Surveillance. Indian Journal of Science and Technology 16(25): 1877-1887. https://doi.org/10.17485/IJST/v16i25.1279

* **Corresponding author**.

ssk.saravanakumar@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

An Enriched Model of Neutrosophical Fuzzy and Grasshopper Convolutional Neural Network Based Moving Object Detection and Classification to Improve Video Surveillance

S Saravanakumar[1]*, M Lingaraj[2]

1 Research Scholar, Research and Development Centre, Bharathiar University, Coimbatore, Tamilnadu, India
2 Associate Prof & Head, Dept of CS, Sankara College of Science and Commerce, Coimbatore, Tamilnadu, India

Abstract

Objectives: To identify a suitable object recognition method for video surveillance systems, especially in traffic monitoring, to track and sense multiple objects and classify them by employing conventional algorithms in order to boost accuracy. **Methods:** Neutrosophical Background Subtraction (NBS) and Grasshopper Optimized Convolutional Neural Network (GO-CNN) are employed to detect and track the objects in real time. The Neutrosophical Fuzzy Background Subtraction Method (NFBSM) is utilized to segment moving objects from the background in terms of truthful degree, false degree, and in-deterministic degree. CNN-based object tracking techniques, along with the grasshopper optimization algorithm (GO), are used for reliable object detection and hyper-parameter fine tuning. The NFBSM+GO-CNN model extracts the features from the dataset (collected from YOLO-V-3 from Kaggle) and selects a suitable function to perform object detection and classification of moving vehicles with a high accuracy rate. Four main features are used such as colour, texture, motion, and shape. PYTHON is used for implementation and comparative analysis, and the findings are compared with baseline models D-CNN and CNN-MOD. **Findings:** The proposed NFBSM+GO-CNN method outperforms with 98.2% accuracy, 96.3% improved precision, and 96.8% F-measure, which is comparatively higher than the baseline models in terms of real-time object detection and classification. **Novelty:** The outcome of NFBSM+GO-CNN clearly shows that the proposed model has the ability to detect the object in real-time and classify the vehicle in road traffic monitoring with a high accuracy rate. The proven results outperform existing models such as D-CNN and CNN-MOD.

Keywords: Object Detection; Classification; Machine Learning; CNN; Optimization; Video Surveillance Systems

1 Introduction

In the Smart Transportation Technology that has been extensively researched over the past several years, vehicle detection plays a crucial role[1]. Detection of moving objects and classifying them forms the primary tasks in any researches of that domain. Most of the existing models focus of identifying single object and classifying them whereas the proposed work focuses on identifying multiple objects in the video frames, classify them and the accuracy using NFBSM+GO-CNN methods. The purpose of this work is to propose a novel method for (i) real-time object detection (ii) object classification (iii) boost the accuracy level by employing NFBSM+GO-CNN model.

As the overall number of vehicles in urban areas is growing quickly, the need for intelligent facilities has also increased. A crucial component of the intelligent transportation system, which comprises the detection, categorization, tracking, and counting of vehicles, is traffic monitoring[2]. Traffic supervisors can use traffic video detection to relieve traffic congestion and enhance highway planning, which is one of its main benefits. Real-time image processing and flexible staged computerized pattern recognition are used to recognize vehicles in videos. Maintaining the proper operation of automated or constantly operating systems depends heavily on real-time processing[3]. The researchers are focusing on video processing which contributes more in moving object tracking like vehicle tracking, detection and classification, to improve the process of traffic monitoring systems. For traffic analysis, video monitoring, and public safety, object tracking is an essential step in computer vision. The two associated elements of video surveillance are object detection and object tracking. Prior to completing challenging tasks like tracking, object identification in videos is the initial stage[4]. A survey on video object detection with advanced research methodology which involves classification of video objects was conducted[5]. The association and discrimination between detection of objects and relevant tasks were discussed in detail. Nearly forty algorithms were considered for conducting the survey. To discover the region of suspected area of fire by extracting the spatial characteristics, a method was designed[6]. The bounding boxes are generated in each frame to extract and accumulate the features using LSTM. The classification is done based on the voting system to categorize the area as a non-fire or fire region. A deep learning model-based monitoring system has been developed[7] to identify the tracked objects' presence. The authors compared the process of object detection using different deep learning models to determine which model achieves the highest inference speed and precision rate. A conventional algorithm for performing object detection was developed[8] which also compares the performance of both conventional and deep learning models. A detailed survey-based face identification, salient object identification and pedestrian identification are discussed in detail. The issues that still exist in deep learning-based object detection is also discussed. A novel generative framework[9] was developed, which is capable of synthesizing photo-realistic face images with ageing effects without paired samples and accomplishes face age advancement and regression in a complete structure, and solving an unsupervised multi-domain translation problem between images. An innovative background subtraction approach based on Bayesian generative adversarial networks and parallel vision is described in[10]. Initially background image is extracted using the median filtering algorithm. In order to improve the background subtraction outcomes in complicated scenarios, parallel vision theory is used after Bayesian GANs is used to categorize every pixel into foreground and background. In[11] the authors developed a convolutional network-based classification model to track the objects. The method is suitable only for single object tacking. The issue of uncertainty is not concentrated in this work to attain reliability in moving object tracking. In[12] the author's devised acceleration based deep neural network which uses temporal dependencies of moving objects to track their movement. With the variation obtained from of moving objects, the objects in video are perceived. A computationally efficient method is employed for high-resolution images[13], which detects moving regions on a resized image while maintaining moving regions on the original image with mapping coordinates. The deep learning-based object detection models are broadly classified into two categories, i.e., one-stage detector and two-stage detector. one-stage detector was experimented using custom CNN model with hyper-parameter tuning[14] to improve object classification and detection accuracy of deep learning models. To eliminate difficulties in tracking and enhance detection, combination of a convolutional neural network (CNN) and a histogram of oriented gradient (HOG) descriptor[15] was used which stored the appearance features in an array and passed them into the network (CNN) for further processing. The difficulty in searching and retrieving persons in multicamera surveillance systems, especially with cluttered backgrounds and variations in appearance among multiple cameras is discussed using instance segmentation[16] It uses attributes such as colour and type of clothes to describe a person. The recent moving object detection methods are evaluated in[17] based on their practical application in detection of seen scenes and unseen scenes

2 Methodology

In this proposed work two different algorithms are developed to improve the moving object detection in uncertainty condition and handling the problem of over-fitting in convolutional neural network for classification of moving objects. The architecture of the enriched model of Deep Convolutional Neural Network based Moving Object Tracking to improve video Surveillance is shown in Figure 1 . The first phase uses the neutrosophic fuzzy subtraction method to discriminate foreground from background, the neutrosophic feature extraction modelling is used to track the moving objects and the grasshopper optimized convolutional neural network is used for moving object classification.

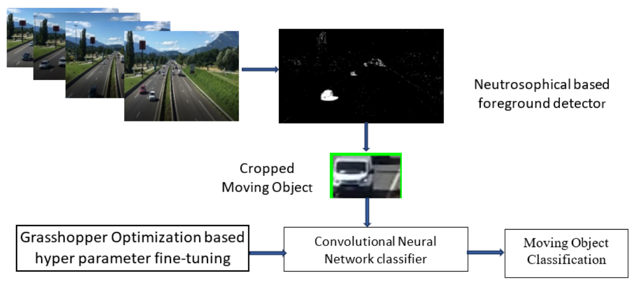

Fig 1. Overall Framework of the proposed model

2.1 Neutrosophic Fuzzy Background SubtractionNFBS Method for Moving Object Detection & Pre-Processing

The working principle and pre-processing data of the proposed NFBS method to detect moving objects in a video by computing successive video frames is shown in below steps. The frame quality must be excellent prior to beginning any video processing procedure. The dataset is collected from Kaggle repository. During this phase, the video is evaluated with noise from salt and pepper, while improving frame quality is considered. These noises have a greater potential to degrade the picture quality of the video. Applying the median filtering technique, results in the raw video frames being denoised.

1. Input Video - Collected from Kaggle
2. Convert into frames
3. Apply Neutrosophical Fuzzy Background Subtraction
4. Converting each pixel of input frame into Neutrosophical values $P_x(i)$ = <$Tr_x(i)$,$FL_x(i)$,$ID_x(i)$>
5. $Tr_R(i)$,$FL_R(i)$,$ID_R(i)$> - <$Tr_C(i)$,$FL_C(i)$,$ID_c(i)$>
6. Compute Threshold
7. Foreground object Segmentation and Detection

2.1.1 Neutrosophic Representation

While the concept of fuzzy sets is very effective at handling uncertainties resulting from the ambiguity or biased belonging of a component in a set, it is unable to model all types of uncertainties that occur in various real-world ailments, including those that involve insufficient data because it focuses on only membership degree μ. Here F is a fuzzy set in the universe information V which is defined mathematically in equation (1)

$$F = \{v, \mu_f(v), | v \in V\} \tag{1}$$

As an alternative to fuzzy when the situation is to handle the insufficient data for defining the impreciseness intuitionistic fuzzy is introduced as shown in equation (2) which holds two grade membership and non-membership with the restriction that two grades sum must be either less or equal to unity.

$$F = \{< v, \mu_f(v), \gamma_f(v) > | v \in V\} \tag{2}$$

But the Neutrosophic Sets (NS) [18] are developed to handle all the uncertainty problem face by real world such as indeterminacy, inconsistency and incompleteness. Thus, neutrosophic set is considered as generalization of multivalued logics, intuitionistic logics, paraconsistent logic and fuzzy set as defined in equation (3).

$$F = \left\{ < v, \widetilde{\mathrm{T}}_f(v), \check{\mathrm{I}}_f(v), \widetilde{\mathrm{FL}}_f(v) >; v \in V, \widetilde{\mathrm{T}}_f, \check{\mathrm{I}}_f, \widetilde{\mathrm{FL}}_f \varepsilon \right]^{-} 0, 1^{+}[\} \tag{3}$$

NS is defined with three unique functions known as truthiness ($T_{f,}$), indeterminacy (I_f)and falsity (FL_f) which are independent membership functions [19].

In this proposed work the neutrosophical concept is implied to handle indeterminacy, inconsistency and impreciseness in moving object detection.

2.1.2 Neutrosophical Background Subtraction method for Moving Object Detection

Here, the stationary and non-stationary objects are discriminated from the video after it is converted to set of frames using neutrosophic background subtraction method. The reference frame and the current frame of video V is represented in terms of neutrosophic values as shown in equation (4) and (5)

$$RF = \{< v.T_{rf}(v), I_{rf}(v), FL_{rf}(v) >; v\, \varepsilon\, V\} \tag{4}$$

$$CF = \{< v.T_{cf}(v), I_{cf}(v), FL_{cf}(v) >; v\, \varepsilon\, V\} \tag{5}$$

Determine the difference among the background and the foreground object using equation (6)

$$NHD(RF,CF) = \frac{1}{3m}\sum_{i=1}^{m}\left(\left(T_{rf}(v)^{i} - T_{cf}(v)^{i}\right| + |I_{rf}(v)^{i} - I_{cf}(v)^{i}| + |FL_{rf}(v)^{i} - FL_{cf}(v)^{i}|\right) \tag{6}$$

Here m is the number of pixels in a frame. In this work the hamming distance is used to discriminate the foreground and the background images.

To classify the foreground object from the background, discrimination process is accomplished by neutrosophic hamming distance modelling[20] and with obtained difference, the result is compared with the threshold value to determine if the input pixel belongs to foreground or background object. The checking process is done as shown below

If $NHD(RF,CF) > Threshold\ Value$ **then**
Pixel in the foreground
Assign '1' to the corresponding foreground vector array
Else
Assign '0' to the corresponding foreground vector array
End If

2.2 Classification of Moving Objects using Grasshopper optimized Convolutional neural network

After extracting the salient features on the background subtracted frames, the classification to reject outliers from the true moving objects is carried out using the grasshopper optimized convolutional neural network (GO-CNN). The salient features extracted through the neutrosophic similarity measure are based on colour, shape, motion and texture. These are passed as input to the deep learning model to train the classifier which discriminates the moving objects from false alarms through grasshopper optimized convolutional neural network (GO-CNN). Deep learning algorithms use a trained neural network to assess whether or not an unseen testing object is a moving target by training the network's weights on a huge dataset. Rectified linear units (ReLU) are used to activate the neurons after applying 16 filters with a 3x3 convolution kernel to produce feature maps as seen in Figure 2. To prevent internal covariate shift during mini-batch optimisation, the batch normalisation unit is introduced between the convolution and ReLU. The size of the feature map is then reduced, and by applying spatial down sampling process known as the max-pooling method, to increase the number of filters for the layer that follows. The convolution of batch normalization and neuron activation is done repeatedly with max pool operations for 2 layers with different filtering operations for32 filters of 3x3x16 and 64 filters of 3x3x32. At last, it deploys a fully connected neural network with a soft-max function to determine the binary classification label. During the classification, it is possible to identify the conspicuous points on the candidates for moving objects by feeding the neural network that has been trained with known format and later tested using new video frame.

Convolution neural networks(CNN)[21,22] can learn complicated objects and patterns and passed into the input layer, output layer, several hidden layers, and numerous parameters. The given input is sub-sampled using convolution, pooling, and an activation function. All of these are hidden layers that are only partially connected, and the final stage is the fully connected layer that yields the output layer. The output keeps its original form and size like the input image.

2.2.1 Convolution function

The technique of combing two functions to create the other function is known as convolution. With the help of filters and convolution, CNN creates a feature map from the input frame.

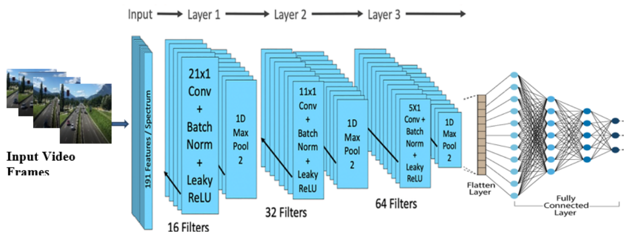

Fig 2. Structure of the proposed convolutional neural network

2.2.2 Kernels/Filters Function

In the network, filters are weights and bias vectors that are created randomly. In CNN, rather than having distinct weights and biases for each neuron, different neurons share identical weights and biases. A large number of filters can be produced, each of which extracts a distinctive property from the input. Kernels are another name for filters.

2.2.3 Convolutional Layer

The main building block of CNN is convolution layer where the process of convolution takes place. Typically, this layer has

- Vector inputs (Image/ Video frames)
- Feature Detector (Filters)
- Feature maps as output vectors

The feature map is generated using dot product of input vector and filters. The fundamental information is preserved as a matrix while this layer detects and extracts the best features and patterns from the input video frame. A feature map, or dot product between combinations of vectors, is created by multiplying the matrix illustrating the input image element by element with filters. The significant features of CNN [22]are regional connection and sharing parametric. In CNN, each neuron is connected with the specific portion of input frame, unlike the conventional neural network wherein each neuron is fully connected. With CNN, a specific filter dimension is selected, and it is slid over those portions of the input frame significant features. CNN uses a number of filters, each of which scans the entire image and understands certain aspects of the input frame. Sharing Parametricis the distribution of weights across every neuron in a specific feature map, each of them shares the same weight amounts.

2.3 Process of Batch Normalization

The input is normalized to decrease the internal co-variate shift to regularize the network and it is done in between the layers of activation and convolution. Higher learning rates are made possible by batch normalization, which can shorten the training times and improve performance. Each layer can learn independently without becoming increasingly reliant on the other layers.

2.3.1 ReLU Layer (Rectified Linear Unit)

The rectifiedlinear unit process is done after the convolution process. To conduct non-linear relationships by neural network, ReLU implements activation function.ReLU changes all negative values in a given matrix (x) to zero while holding all other values constant.When the input layer of CNN receives the normalized images or frames with same size, the convolution layer will identify the neuron's output, which is connected to the local regions of the input, by the computation of scalar product among their weights and the connected region to the input. The rectified unit (ReLu) is used for applying activation function based on element-wise like sigmoid to the output of the activation generated by the preceding layer.

A single pixel in the feature map of convolutional layer is computed as in equation (7)

$$\mathrm{CLm} = af\,(\mathrm{f}_{\mathrm{i}}^{*}\omega + \beta) \tag{7}$$

Where $_{CLmis}$ the m^{th} pixel value in the feature map, pixel value in the local receptive field related to $_{CLmi}$is represented as f, w represents weight and β denotes bias are the coefficients determined using feature map and the af represents the activation function.

To reduce the spatial volume and hyperparameters of activation function, the pooling layer is used as down sampling process. After the pooling layer, is the fully connecter layer, which performs the process of neural network that attempts to generate the category scores from the activation for classification of moving objects. The pooling layer subsamples the feature map. For instance, in pooling layer the feature maps k^{th} unit is calculated using the equation (8) given below

$$\mathrm{PLm} = FM\left(\gamma^{*}\mathrm{DN}(\mathrm{CL}) + v\right) \tag{8}$$

In pooling layer, the feature maps m^{th} unit value is denoted as $_{PLm,,}$CL is the feature map's vector value in convolution layer. The coefficients and bias are represented using γandn.

Down is the function of sub sampling. To perform sub sampling, max pooling function is applied and thus the DN(CL) is computed as in equation (9)

$$\mathrm{DN}(CL) = \max\left\{CL_{\mathrm{r},1} \mid \mathrm{r} \leq \mathrm{s}, \mathrm{l} \leq \mathrm{s}, \mathrm{r}, \mathrm{l} \in \mathrm{z}^{+}\right\} \tag{9}$$

where, $\mathrm{CLr}_{,l}$ is the pixel value of the feature map in unit CL and r is the sub sampling size. The convolutional and pooling layer will acquire low-level information of the input, and the high-level feature extraction is obtained from the stack of all convolution and pooling layer. The output layer is linked to the formal layer, which consist of soft-max regression. The soft-max performs the classification of given input frames into different objects using non-normalized output to a probability distribution. The output of the fully connected layer is passed to the soft-max layer and the outcomes are converted to probability. The sum of multi-class probability values is equal to 1. From the upper layer the output vector VT, Classification of class C based on the probability is defined as shown in equation (10)

$$\mathrm{Pb}\left(\mathrm{DN}^{(\mathrm{VT})}\right) = (\mathrm{CL} \mid \mathrm{VT}, \theta) = \frac{E^{\varphi^{T}_{CLVT}}}{\sum_{n=1}^{M} E^{\varphi^{T}_{CL}}}$$

Where, group identity of VT is denoted as $\mathrm{DN}^{(\mathrm{VT}),}$ M is the number of categories and j is the weight vector factor.

2.4 Hyperparameter Optimization in CNN

In the fully connected layer, the feature vector is used to classify the moving object among different types of moving objects in the video. All the output of the current layer is connected as the input to the next layer. As all the parameters are occupied in fully connected layer that result in overfitting problem in the conventional CNN, dropout method is used to overcome the overfitting issue in moving object detection and classification. In this proposed work the problem of overfitting problem in the fully connected layer which is the essential component of CNN, is handled by adopting the knowledge of grasshopper optimization algorithm. Based on the fitness evaluation, the best fittest hoppers' values are assigned to the hyperparameters of CNN

2.4.1 Grasshopper Optimized CNN for hyper-parameter optimization

In this proposed work the unique characteristic of grasshopper [23] in search of their food is used to fine tune the hyperparameters of CNN to optimize the performance of the moving object classification. Grasshoppers are a type of bug that some find bothersome. These insects are known as pests in the agricultural industry because they frequently cause damage to the crop. In nature, grasshoppers may occasionally live alone but almost always congregate in huge swarms. When the size of the swarm is excessive, it's always a farmer's worst nightmare. Grasshoppers exhibit distinct traits of swarm behaviour both as adults and as nymphs. Nymph grasshoppers are huge in numbers and move like roller cylinders [23]. They nearly consume all of the nearby plants as they march forward. When the nymphs reach adulthood, they form an airborne swarm and migrate over great distances. The swarm moves very slowly during the larval phase, which is a distinctive trait of grasshoppers at this period. In contrast to this key characteristic, the swarm moves abruptly as its ages.

Procedure for fine-tuning hyperparameters of CNN using Grasshopper Optimization algorithm

- Initialize the hyper parameter values to swarm Gw_i {i = 1,2,3..m)
- Assign cm, cn, max_itr
- Calculate each agent fitness value
- BS $\rightarrow$Best Search agent
- While T<max_itr
- Modify C = $c_m - ct\,\frac{cm-cn}{MX}$

Where MXis the maximum number of iterations, $c_n$denotes minimum value, cm is the maximum value and t indicates the current iteration.

- For each search agent
- Perform normalization on the distance between artificial grasshoppers

• Update the position of search agents using the formula as follows

$$SG_i^d = \left(\sum_{j=i=1}^{N} dc \frac{UB_d - LB_d}{2} s(|sg_j^d - sg_i^d|) \frac{sg_{j-sa_i}}{d_{ij}} \right) + \overline{BS_d}$$

Where UB_d is the dth dimensions upper bound and lower bound is signified as LB_d,BS_d best solution established so far. dc is a decreasing coefficient to contract the extravagance, zone of attraction and repulsion.

• Reverse the direction of the current search agent if it is heading outside of the border areas.
• End
• Update BS if there is improved search agent found
• T = T + 1
• End

2.5 Algorithm

Enriched model of Neutrosophical Background Subtraction method with Grasshopper optimization integrated Convolutional Neural Network based Moving Object Tracking to improve video Surveillance

Step 1: Obtain the original video and convert into frame

Step 2: Apply preprocessing and neutrosophic background subtraction to detecting moving object

Step 3: Pass the input frames into 3D convolution network to receive the significant features based on spatio-temporal evidence

a) To extract the motion data as input for this work, the CNN used 11 frame cubes.

b) The frame size is reduced from 160x120 to 120x120 in order to use less memory and simplify processing.

c) The CNN, which has 120 * 120 * 11 inputs and varies the number of feature maps and kernel sizes for each layer.

d) The first convolutional layer uses kernels with a size of 3x3x16, the second uses filters with a size of 3x3x32, and the third uses kernels with a size of 3x3x64.

e) To combat the issue of overfitting, the kernel size for the layers of max pooling is 2 * 2, which gradually reduces the spatial information. · By utilizing grass hopper optimization, you may find the best feature vectors for classifying moving objects.

f) As a final step, the CNN's fully connected layer converts all of the activation data from the previous layer into 6400 feature vectors.

g) The soft-max layer is comprised of the output units that combine the results of categories of moving objects in video.

Step 4: The class to which the chosen object belongs is outputted once the feature vectors have been passed to the CNN's fully connected layer.

Result: Moving objects Classification

3 Results and Discussions

The proposed work NBFSM based moving object detection and grasshopper optimized convolutional neural network (GO-CNN) for moving object classification is deployed using python code. This section discusses in detail about the simulation analysis of the proposed model (NFBSM+GO-CNN). The object detection from road traffic video monitoring dataset is collected from the Kaggle repository[24]. The performance of the proposed NFBSM+GO-CNN is compared with existing conventional object detection algorithms. The metrics used for comparison are precision, recall, f-measure and accuracy.

Each pixel must make a choice that can fall into one of these four categories, whether it is correct or incorrect.

• True positive(TRP): Correct verdict as Non-Stationary object
• False positive (FLP): Incorrect verdict as non-Stationary object
• True Negative (TRN): Correct verdict as Stationary object
• False Negative (FLN): Incorrect Verdict as background

Accuracy which is the base metric is the number of correct predictions from all predictions made and is calculated as shown in the below equation.

Accuracy:

$$Accuracy : AC = \frac{TRP + TRN}{(TRP + TRN + FLP + FLN)} \tag{11}$$

Precision, as known as positive predictive value is the measure of no. of many of the positive predictions made are correct which is calculated as shownin the below equation.

Precision

$$Precision : PR = \frac{TRP}{(TRP + FLN)} \tag{12}$$

Recall, also referred as Sensitivity is the measure of the no. of the positive cases predicted correctly over the total positive cases in the data, calculated as shown in the below equation.

Recall

$$RL = \frac{TRP}{(TRP + FLP)} \tag{13}$$

F-measure is stated as the harmonic mean of precision and recall and is calculated as shown in the below equation.

F-Measure

$$FM = 2 * \frac{PR * RL}{(PR + RL)} \tag{14}$$

Figure 3 illustrates the accuracy of detecting objects in video by three different algorithms for frame index varying from 0 to 30. The results explored that the proposed Neutrosophic fuzzy background subtraction method detects the moving objects more accurately compared to the conventional BSM and FM. The proposed NFBSM computes each object's characteristics in the form of triplet representation towards the degree of truthiness as an object, degree of falsity as an object and degree of indeterminacy as an object. Based on these three independent factors, the proposed NFBSM detects the foreground objects more accurately by tackling indeterminacy in avoiding non-stationery background objects.

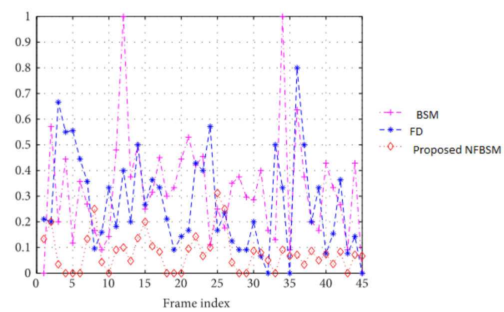

Fig 3. Accuracy Analysis

The false alarm rate of conventional background subtraction method, frame difference method and the proposed Neutrosophical fuzzy background subtraction method is displayed in Figure 4 . While comparing with the other two algorithms, the proposed NFBSM produces less False Alarm Rate (FAR), because the noise present in the frame spoils its quality and extracting the information from frame becomes difficult. Hence, to overcome this problem the NFBSM represents the objects in terms of grade of belongingness, non-belongingness and indeterminacy to handle the complex situation in moving object detection especially in presence of noise, illumination and shadow in a video.

Figure 5 shows the comparison of precision rate obtained by three object classification models. After the object is detected, classification of objects is an important task in video surveillance of traffic monitoring system. For classification of moving objects, the input of the foreground objects colour, shape and texture are in the neutrosophical representation. The input of the proposed GO-CNN is in neutrosophical values and the convolutional neural network with its multiple layers detects the important features. The hyperparameters involved in classification of the objects are fine-tuned usinggrasshopper optimization method. Thus, the proposed GO-CNN produced highest precision rate compared to the other existing DNN and CNN models.

The proposed GO-CNN achieves highest recall values compared to other two algorithms DNN and CNN shown in Figure 6. The GO-CNN, with its ability of optimizing the hyper-parameter values using grasshopper food foraging behaviour the values are fine-tuned. The fitness value of each grasshopper based on its food source is computed and the best grasshoppers are selected for assigning the learning rate and weight parameters values in CNN. Thus, the improvement done in training the dataset empowers the process of classification of moving objects. The uncertainty in moving objects' colour, shape and texture characteristics are defined by the neutrosophical values based on pixel-by-pixel computation. Hence, the proposed GO-CNN accomplishes highest recall rate compared with the conventional DNN and CNN.

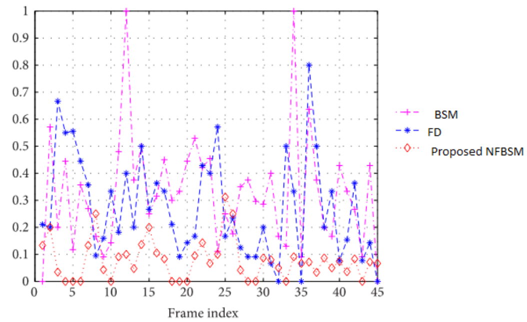

Fig 4. False Alarm Rate

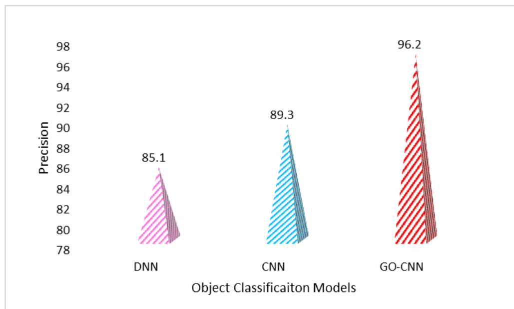

Fig 5. Precision Analysis

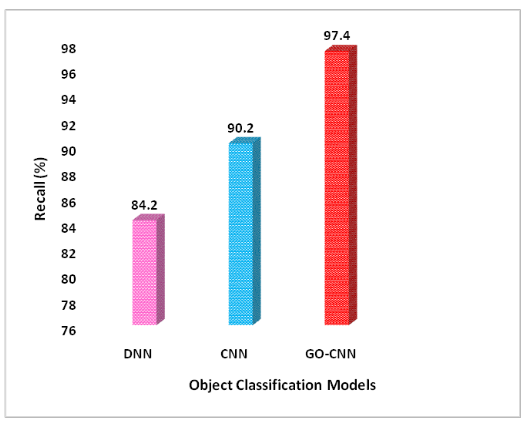

Fig 6. Recall Analysis

The f-measure value shown in Figures 7 and 2 and depends on the precision and recall values and it also proves the efficiency of the proposed GO-CNN compared with CNN and DNN. In the conventional CNN and DNN the hyper-parameter values are assigned based on the trial-and-error basis and using gradient descent method.

But proposed GO-CNN integrates the knowledge of grasshopper optimization to fine tune the parameters involved in detection and classification of moving objects and the neutrosophic values are used for tracking the moving objects based on their characteristic motion, shape and texture. Thus, the challenges in moving object tracking in video surveillance is overcome by adopting the two novel models introduced in this work.

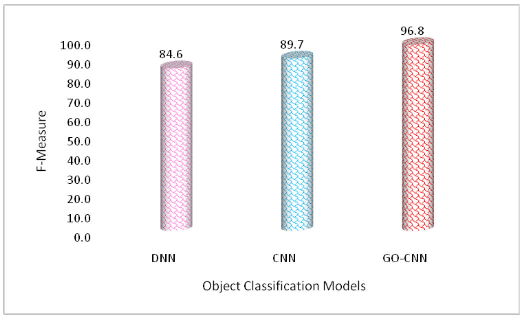

Fig 7. F-Measure Analysis

4 Conclusion

This proposed study mainly focuses on real-time object detection and classification of moving objects for traffic monitoring and video surveillance. Neutrosophical Background Subtraction (NBS) and Grasshopper Optimized Convolutional Neural Network (GO-CNN) are the two novel ML methods employed to boost the accuracy of real-time object detection and classification. In the first stage, moving objects are detected by discriminating between foreground and background objects by introducing Neutrosophical fuzzy-based movement detection (NFBSM). Each pixel in the entire frame is represented using the triple format of Neutrosophical values to handle inconsistency due to the dynamic spatial and temporal elements of video. The Neutrosophical model determines the colour similarity, shape similarity, and texture similarity among objects. These extracted features are used in the second stage as input to the convolutional neural network. During the training phase of the CNN, the accuracy rate is optimized by fine-tuning the hyperparameters involved in the classification process. YOLO-V-3 from the Kaggle dataset is utilized. The simulation results proved that the proposed algorithm NFBSM+GO-CNN achieves the highest moving object detection and classification rate with 98.2% accuracy, 96.3% improved precision, and 96.8% F-measure. A few limitations of this method are that only four main features - colour, texture, motion, and shape are used. In the future, this method may be enhanced by adding other features like patterns, visuals, etc. Other noise removal algorithms can be employed to enhance the images, and the same can be implemented for other types of videos as well.

References

1) Jiwan D, Chu S, Hoi H, Pengchengwu J, Zhu Y, Zhang J, et al. Deep learning for content-based image retrieval: A comprehensive study. *Proceedings of the ACM International Conference on Multimedia*. 2014;p. 157–166. Available from: https://doi.org/10.1145/2647868.2654948.
2) Bouwmans T, Sobral A, Javed S, Jung SK, Zahzah EH. Decomposition into low-rank plus additive matrices for background/foreground separation: A review for a comparative evaluation with a large-scale dataset. *Computer Science Review*. 2017;23:1–71. Available from: https://doi.org/10.1016/j.cosrev.2016.11.001.
3) Nagarkar P, Candan KS. An index structure for efficient execution of set queries in high dimensional spaces. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 2018;p. 477–486. Available from: https://doi.org/10.1145/3269206.3271691.
4) Zhang Y, Li X, Zhang Z, Wu F, Zhao L. Deep learning driven block wise moving object detection with binary scene modelling. *Neurocomputing*. 2015;168:454–463. Available from: https://doi.org/10.1016/j.neucom.2015.05.082.
5) Licheng J, Ruohan Z, Fang L, Shuyuan Y, Biao H, Lingling L, et al. New Generation Deep Learning for Video Object Detection: A Survey. *IEEE Transactions on Neural Networks and Learning Systems*;33:3195–3215. Available from: https://doi.org/10.1109/TNNLS.2021.3053249.
6) Kim B, Lee J. A Video-Based Fire Detection Using Deep Learning Models. *Applied Sciences*. 2019;9(14):2862. Available from: https://doi.org/10.3390/app9142862.
7) Boukabous M, Azizi M. Image and video-based crime prediction using object detection and deep learning. *Bulletin of Electrical Engineering and Informatics*;12(3):1630–1638. Available from: https://doi.org/10.11591/eei.v12i3.5157.
8) Kumar K, Kumar K, Gupta CLP. Object Detection in Video Frames using Deep Learning. *International Journal of Computer Applications*. 2022;183(51):33–39. Available from: https://www.ijcaonline.org/archives/volume183/number51/kumar-2022-ijca-921930.pdf.
9) Zeng J, Ma X, Zhou K. PhotoRealistic Face Age Progression/Regression Using a Single Generative Adversarial Network. *Neurocomputing*. 2019;p. 1–16. Available from: https://doi.org/10.1016/j.neucom.2019.07.085.
10) Zheng W, Wang K, Wang FY. A novel background subtraction algorithm based on parallel vision and Bayesian GANs. *Neurocomputing*. 2020;394:178–200. Available from: https://doi.org/10.1016/j.neucom.2019.04.088.
11) Lee DH. CNN-based single object detection and tracking in videos and its application to drone detection. *Multimedia Tools and Applications*. 2021;80(26-27):34237–34248. Available from: https://doi.org/10.1007/s11042-020-09924-0.
12) Yoo JY, Ko JH. Acceleration of DNN-Based Video Object Detection Using Temporal Dependency of the Object Size. *2021 International Conference on Information and Communication Technology Convergence (ICTC)*. 2021;2021:1182–1184. Available from: https://doi.org/10.1109/ICTC52510.2021.9620830.

13) Zhu H, Wei H, Li B, Yuan X, Kehtarnavaz N. Real-Time Moving Object Detection in High-Resolution Video Sensing. *Sensors*;20(12):3591. Available from: https://doi.org/10.3390/s20123591.
14) Yadav S, Chaware SM. Video Object Detection with an Improved Classification Approach. In: Bruckstein, M A, editors. Data Management, Analytics and Innovation;vol. 662. Springer Nature Singapore. 2023;p. 511–523. Available from: https://doi.org/10.1007/978-981-99-1414-2_38.
15) Kalake L, Dong Y, Wan W, Hou L. Enhancing Detection Quality Rate with a Combined HOG and CNN for Real-Time Multiple Object Tracking across Non-Overlapping Multiple Cameras. *Sensors*. 2022;22(6):2123. Available from: https://doi.org/10.3390/s22062123.
16) Tseng CH, Hsieh CC, Jwo DJ, Wu JH, Sheu RK, Chen LC. Person Retrieval in Video Surveillance Using Deep Learning–Based Instance Segmentation. *Journal of Sensors*. 2021;2021:1–12. Available from: https://doi.org/10.1155/2021/9566628.
17) Zhao X, Wang G, He Z, Jiang H. A survey of moving object detection methods: A practical perspective. *Neurocomputing*. 2022;503:28–48. Available from: https://doi.org/10.1016/j.neucom.2022.06.104.
18) Salama A, Smarandache F, Yasser I. Neutrosophic Knowledge. vol. 1. 2020. Available from: https://fs.unm.edu/NK/.
19) Mehmood A, Nadeem F, Nordo G, Zamir M, Park C, Kalsoom H, et al. Generalized Neutrosophic Separation Axioms in Neutrosophic Soft Topological Spaces. 2020. Available from: https://digitalrepository.unm.edu/cgi/viewcontent.cgi?article=1476&context=nss_journal.
20) Hwan PC. Neutrosophic ideal of Subtraction Algebras. *Neutrosophic Sets and Systems*;24:36–45. Available from: https://digitalrepository.unm.edu/nss_journal/vol24/iss1/5/.
21) Babaeem, Dinh DT, Rigoll G. A deep convolutional neural network for video sequence background subtraction. 2018. Available from: https://doi.org/10.48550/arXiv.1702.01731.
22) Nandhini TJ, Thinakaran K. CNN Based Moving Object Detection from Surveillance Video in Comparison with GMM. *2022 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI)*. 2022;2022:1–6. Available from: https://doi.org/10.1109/ICDSAAI55433.2022.10028909.
23) Meraihi Y, Gabis AB, Mirjalili S, Ramdane-Cherif A. Grasshopper Optimization Algorithm: Theory, Variants, and Applications. *IEEE Access*. 2021;9:50001–50024. Available from: https://doi.org/10.1109/ACCESS.2021.3067597.
24) Dataset Available From. . Available from: www.kaggle.com/datasets.