Received: May 1, 2021. Revised: July 7 2021.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A Feature Extraction Approach for Multi-Object Detection Using HoG and LTP

Eisha Akanksha[1]* **Pamarthi Rama Koteswara Rao[2]**

[1]*Department of Electronics & Communication, CMR Institute of Technology, Bengaluru 560037, India*
[2]*Department of Electronics and Communications, NRI Institute of Technology, Pothavarappadu, Krishna Dt, India*
* Corresponding author's Email: Eisha.a@cmrit.ac.in

Abstract: In the field of computer vision, object detection is getting more attention due to its huge applications in visual monitoring. Multiple object detection identifies the position of objects or regions of objects in the image or videos. Many methods were developed for detecting multiple objects, but the overall detection accuracy of those methods was limited due to the congested environment, complex background, and similarities between the objects. To solve such an issue, this research study proposed the feature extraction method for multiple object detection using Histogram of Oriented Gradient (HOG) with Local Ternary Pattern (LTP). The Caltech 101 dataset is used in the proposed method where the images are converted to LAB. The process of feature extraction takes place by using the proposed HoG and LTP to detect prominent regions from the image. Further, the obtained features are fused by using Deep Convolutional Neural Network (D-CNN) and then forwarded to Region-based Convolutional Neural Network (R-CNN) to detect the multiple objects. The proposed HoG and LTP feature extraction method has the advantages of improving the classification accuracy by effectively extracting the oriented features and texture features. The proposed method achieved better accuracy of 92.48%, whereas the existing Multi-Object Detection and Tracking (MODT) method achieved an accuracy of 76.23% for the detection of multiple objects.

Keywords: Computer vision, Deep convolutional neural network, Local ternary pattern, Object detection, Region-based convolutional neural network.

1. Introduction

Object detection is an important extensive research issue in the field of computer vision with an enormous range of applications like autonomous driving, advanced driving assistant system, robotic visions, augmented reality, etc. [1]. The main task of object detection is to identify the category and position or regions of specific objects in images and videos. Generally, it is considered as a necessary step to narrow down the object related to the vision process like visual tracking, re-identification of person, and segmentation of semantics [2]. The object detection method utilizes different types of shape patterns for the evidence to identify interesting objects in images or videos. The object identification models are trained with the patterns of shapes which present a similar category of objects to differentiate among various categories. But, it is difficult for a system to identify every appearance of an object accurately, because the features of fundamental object shapes, object poses, and angles of view vary greatly [3]. Multiple object detection aims to identify the trajectory of objects based on similarities between the sequence of images or videos. Initially, target objects are detected in multiple object detection and track the algorithm to evaluate the trajectory of objects by utilizing the outcome of detection [4]. The multiple object detection with tracking makes use of associated data of existing track and new identification from each frame, which forms the trajectory of multiple objects. So, the outcome of data association produces series of detection with unique identities [5].

Multiple object identification is a challenging process of objects that include similar appearances. In this scenario, the objects in motion are the cue for discriminating and tracking the various objects. When single moving cameras are utilized, the

observable cues of motion are contaminated by the movement of the global camera which is not detectable [6]. Object detection proposed a challenge due to degraded qualities of images such as blurriness in motion and defocus on videos which leads to unstable classification for similar objects [7]. The many object detection method has been developed based on deep learning like Convolutional Neural Network (CNN) [8], faster region-based CNN [9], spatial pyramid pooling network [10], region-based Fully CNN [11], You Only Look Once (YOLO) [12], Feature Pyramid Network (FPN) [13]. The existing techniques detect the objects effectively in certain labeled images which required assigning positions and classes of objects and background distributors. However, the assignment process was laborious and time-consuming when the objects were annotated manually [14]. The existing sliding window object detection method suffered from drawbacks such as the handcrafted features include limited representational power to detect the objects accurately. Further, CNN achieved success in object detection by improving the performance over the traditional approach. But, due to challenging environments such as object occlusion, larger variation in object scale, and poor light conditions the CNN detector was not achieved good accuracy.[15]. To solve such an issue, a proposed feature extraction-based method using HoG and LTP for multi-object detection. The CALTECH101 dataset is utilized in the proposed HoG and LTP method which is subjected for RGB to LAB Conversion. Then, the features are extracted from LAB-converted images using HoG and LTP . The proposed HoG and LTP method effectively extracts the oriented features and texture features from LAB converted images thereby improves the accuracy of classification. The proposed HoG and LTP method identifies the similar appearance objects from the images effectively. The prominent features from HoG and LTP are fused into one matrix by utilizing DCNN and forwarded to R-CNN to detect multiple objects.

The paper is organized as follows, the survey of existing techniques based on object detection is reviewed in Section 2, proposed feature extraction method by using HoG and LTP for multi-object detection is explained in Section 3. The experimental results and discussion is described in Section 4 and the conclusion of the proposed method is present in Section 4.

2. Literature review

The existing techniques based on object detection with their advantages and disadvantages are reviewed and described in this section.

Elhoseny [16] developed a MODT model for the video surveillance system. The MODT method utilized the optimal Kalman filtering method to track the motion of an object in a video frame. The videos were converted into morphological operations based on the total number of frames that utilized the growing region of the model. The Kalman filtering was applied after separating the objects by using a probability-based grasshopper approach to optimize the parameters of the objects that detect in every frame with similar calculation. The MODT method identified the objects that were moving in images without noise and under a lower level of illumination. However, the Kalman filter utilized showed severe occlusion in objects which reduced the rate of detection.

Fu et.al. [17] developed a region-based Convolutional Neural Network Framework for arbitrary and multi-scale object identification in remote sensing images. The feature fusion architecture was developed to extract the features of detection in Region of Interest (RoI) wise. The Oriented Region Proposal Network (RPN-O) was established to get the accurate position of arbitrary oriented objects and adopted RoI pooling to avoid the orientation changes. The developed CNN framework was robust to objects in the remote sensing images added the anchors with extra scales and angles for RPN to improve in detection. But, the developed feature fusion method was not able to identify similar appearances and suffered from detecting the background of images which reduced the performance of object detection.

Rashid et al. [18] developed a fusion strategy, joint selection based on deep CNN and Scale-Invariant Features Transform (SIFT) for object detection and classification. Initially, it implemented the improved salient approach to get the point features and deep CNN features from the two models like AlexNet and VGG. To select the effective features, the reyni entropy controlled technique was used based on deep CNN pooling and point matrix of SIFT. At last, the selected features are combined with the matrix in a series which was transferred to an ensemble classifier. The developed method was automatically detected with the labeled objects from a larger number of image data which reduced human intervention. However, the reyni entropy method showed a problem in selecting the features because the size of inputs was different for every model.

Huang et.al. [19] developed dense connection and spatial pyramid pooling based YOLO method for the detection of an object. The dense connection method is utilized to optimize the connection of the backbone

network structure. The improved spatial pyramid pooling was established to be concatenated to pool the regional feature of various scales in similar convolutional layers with lesser location error. The loss functions such as mean square error loss and cross-entropy were used for training as well as detection of objects. The detection rate was improved by strengthening the propagation of features and provided various information flow in the network. However, the developed method has not augmented the dataset properly and showed lesser accuracy in detecting the objects due to large-scale variance, complex environments, and rotational variations.

Cai and Vasconcelos [20] developed a cascade Region-based CNN method for a higher quality of object detection and segmentation. The cascade is applied for the inference to remove the mismatched quality of detectors and hypotheses. The resampling approach significantly improves the quality of the hypothesis, which provided the positive training set with similar sizes for every detector and minimizes the overfitting problem. However, the developed method maximized the diversity of samples that are used to predict the masked object as the segmentation process was a patch-based operation that includes a larger number of highly overlapping instances.

3. Proposed methodology

In this research, proposed a feature extraction method by using HoG and LTP for multi-object detection. The caltech101 dataset is used in the proposed method for multi-object detection. The Red Green Blue (RGB) color space of the dataset is converted into the LAB. The process of feature extraction takes place by using Histogram of Oriented Gradient (HOG) to detect salient regions from the image and the obtained features are fused by using D-CNN. The R-CNN is used in the proposed method to classify the feature vectors obtained from the process of fusion to detect the objects. The block diagram of the proposed feature extraction method using HoG and LTP for multi-object detection is shown in Fig. 1.

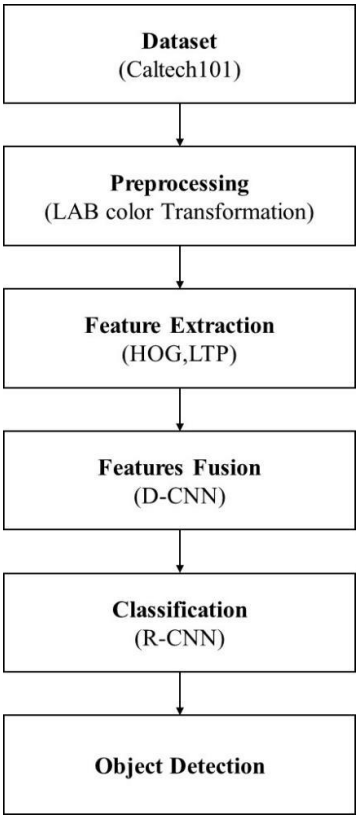

Figure. 1 The block diagram of the proposed feature extraction method by using HoG and LTP for object detection.

3.1 Dataset

The caltech101 dataset is used in the proposed method for multi-object detection. The Caltech 101 dataset is the digital images created in September 2003 at the California Institute of technology which was compiled by Fei-Fei Li. The Caltech 101 includes 9146 images which are split between 101 different categories of objects such as watches, faces, ants, piano, etc. and includes background or clutter categories. The categories of every object includebetween 40 to 800 images where the category of faces utilizes more images than another category. Every image is of the pixels 300×200, the images of oriented objects like motorcycles and aeroplane are mirrored from left to right-aligned and the vertically oriented images like buildings are rotated on the off-axis. The images are provided with a set of annotations that describes the outlines of images with script in Matlab for viewing. The sample images of the CALTECH101 dataset are shown in Fig. 2.

3.2 Preprocessing:

The Red Green Blue (RGB) colour space of the dataset is converted into LAB to make the colours in the image look more vibrant which helps in extracting the features effectively. Where L in LAB stands for lightness or luminance, A and B are the chromatic components of colour like RGB and yellow. In the proposed method, the LAB colour conversion identifies the colours in three dimensions which consist as L denotes luminance, A and B denotes the components of colour like green-red and blue-yellow correspondingly. The L channel is bright white with the value of 100, dark black with the value of 0 and

Figure. 2 The sample images of the CALTECH101 dataset

the channels of A, B shows the natural value of RGB image [13].

Let the input of RGB image be $U(i,j)$ of length $M \times N$, converts RGB to LAB, initially RGB to XYZ conversion is carried out to obtain various RGB color space in a similar way. The XYZ color space describes every color that are visible to humans. In XYZ color space, X represents color chromaticity, Y represents the luminance and Z represents the color blue. XYZ conversion takes place by using Eq. (1).

$$\begin{bmatrix}\varphi(X)\\ \varphi(Y)\\ \varphi(Z)\end{bmatrix} = [M \times N] \begin{bmatrix}\varphi^r\\ \varphi^g\\ \varphi^b\end{bmatrix} \quad (1)$$

Where, $\varphi(X)$ is the channel of X extracted from red $\varphi(X)$, $\varphi(Y)$ is the channel of Y extracted from green $\varphi(Y)$ and $\varphi(Z)$ is the channel of Z extracted from blue $\varphi(Z)$. The red, green and blue channels are defined as shown in Eqs. (2-4).

$$\varphi^r = \sum_{k=1} \frac{\varphi^k}{\Delta_k}, k = Red \quad (2)$$

$$\varphi^g = \sum_{k=1} \frac{\varphi^k}{\Delta_k}, k = Green \quad (3)$$

$$\varphi^b = \sum_{k=1} \frac{\varphi^k}{\Delta_k}, k = Blue \quad (4)$$

Where, φ^r, φ^g and φ^b are the channels of red, green and blue. k is the coefficient that depends on the illuminants of a channel. The coefficient of k depends on illuminants and varies from 70 to 175. The conversion of LAB is defined from Eqs. (5-8).

$$\varphi^{*L} = \beta_1(f_y - 16), \beta_1 = 116 \quad (5)$$

$$\varphi^{*A} = \beta_2(f_x - f_y), \beta_2 = 500 \quad (6)$$

$$\varphi^{*B} = \beta_3(f_y - f_z), \beta_3 = 200 \quad (7)$$

$$\varphi^{(L*A*B)} = (\varphi^{*L} + \varphi^{*A} + \varphi^{*B}) \quad (8)$$

where, $\varphi^{(L*A*B)}$ is the LAB converted image, $\varphi^{*L}, \varphi^{*A}, \varphi^{*B}$ are the channels of light and chromatic components, $\beta_1, \beta_2, \beta_3$ are the standard colorimetric values, The functions f_x, f_y and f_z are considered in two ways to prevent infinite loop such as $x = 0$. The function is assumed linear below $x = x_r$ and assumed to match the $\sqrt[3]{x_r}$ part for the function at x_r in both value and slope. The linear functions f_x, f_y and f_z are calculated as shown in Eqs. (8-10).

$$f_x = \left\{\sqrt[3]{x_r} \frac{kx_r+16}{116}, \rightarrow x_r > \in \left| otherwise \right\}, x_r = \frac{x}{x_r} \quad (8)$$

$$f_y = \left\{\sqrt[3]{y_r} \frac{ky_r+16}{116}, \rightarrow y_r > \in \left| otherwise \right\}, y_r = \frac{y}{y_r} \quad (9)$$

$$f_z = \left\{\sqrt[3]{z_r} \frac{kz_r+16}{116}, \rightarrow z_r > \in \left| otherwise \right\}, z_r = \frac{z}{z_r} \quad (10)$$

Where, x, y, z are the color stimulus, x_r, y_r, z_r are the specific white achromatic reference illuminants. The obtained LAB images are forwarded for the process of feature extraction to extract the oriented and texture features which are effective in detecting the multiple objects from images.

3.3 Feature extraction

After converting the images from RGB to LAB, the process of feature extraction takes place by using Histogram of Oriented Gradient (HOG) and Local Ternary Pattern (LTP) to extract the oriented features and texture features from LAB converted images, which improves the accuracy of detection.

3.3.1. Histogram of oriented gradient (HOG)

The histogram counts the number of occurrences with gradients orientation in the region of local spatial of an image which is called a cell. By extracting the features of HOG, the gradients of images are calculated by creating the histogram of orientation at every cell. The histograms obtained from each cell are normalized that gives the HOG descriptor of particular blocks. The steps involved in the HOG feature extraction are described below.

Initially, the LAB frames are converted to grayscale that is followed by computation of gradients by convoluting the images into horizontal and vertical mask such as $[-1\ 0\ 1]\ and\ [-1\ 0\ 1]^T$. The gradients of horizontal and vertical masks are represented as shown in Eqs. (11, 12).

$$G_x(x,y,z) = [-1\ 0\ 1] * \varphi^{(L*A*B)} \tag{11}$$

$$G_y(x,y,z) = [-1\ 0\ 1]^T \times \varphi^{(L*A*B)} \tag{12}$$

Where, $G_x(x,y,z)$, $G_y(x,y,z)$ are the gradients of vertical and horizontal masks, $\times$ is the convolution, $\varphi^{(L*A*B)}$ is the preprocessed image obtained from converting RGB to LAB. the θ orientation at every pixel is computed using a ratio of gradients in horizontal and vertical directions which is shown in Eq. (13).

$$\theta(x,y) = \arctan\frac{G_x(x,y,z)}{G_y(x,y,z)} \tag{13}$$

Further, every block is divided into $M \times N$ cells, where $M \leq N$. For every cell, pixels will be calculated by weighted vote which are the gradients of magnitude at every pixel and the votes were accumulated in the bins of orientation. The bins L are spaced among $o^0\ to\ 180^0$ for the unsigned gradients or 0^0 to 360^0 for signed gradients and L_{th} value of bins which is shown in Eq. (14).

$$\Omega_l(x,y,z) = \begin{cases} G(x,y,z) & if\ \theta(x,y) \in bin_L \\ 0 & otherwise \end{cases} \tag{14}$$

Where,

$$G(x,y,z) = \sqrt{G_x(x,y,z)^2 + G_y(x,y,z)^2}$$

it is the magnitude of gradients at the pixels (x,y,z). The HOG features are extracted from every cell as the generated features that are equal to a number of bins and features are normalized as shown in Eq. (15).

$$N_f = \frac{(\sum_{(x,y,z)\in Block} G(x,y,z)+\epsilon)}{(\sum_{(x,y,z)\in Block} G(x,y,z)+\epsilon)} \tag{15}$$

where, N_f is normalized histogram features, the normalized features of every cell in the block from dimension vector are equal to the product numbers of oriented bins and overall cells in a block which is a final descriptor of a block.

3.3.2. Local ternary patterns (LTP)

The LTP has a three-valued texture operator with efficient and simple descriptors for describing the features. The image pixels are labelled by determining the threshold in the specific neighbor of every pixel with a centred value multiplied by the power of 2 and added to generate a new label or value for the centred pixel. The LTP is considered as an extension of LBP, instead of thresholding based on centred pixel values of neighbor. The threshold t will be defined as pixel values within the range of $-t$ to $+t$ which is considered to assign the value of zero to pixels. The value of 1 will be assigned to the pixels if the value is higher than the threshold value and the value of -1 is lesser than the centred pixels value. The evaluation of the LTP operator is explained in Eq. (16).

$$LTP(i) = \begin{cases} 1 & if\ p_i - p_c \geq t \\ 0 & if\ |p_i - p_c| < t \\ -1 & if\ p_i - pc\ \leq -t \end{cases} \tag{16}$$

Where, t is the user-specified threshold, p_i is the pixel values in neighbor and p_c is the central pixel value. The LTP obtains the texture operators which are less sensitive to noise because it will not be based on the value of centered pixels and not strictly invariant to transformations of gray level. The LTP value is divided among two sub-LBP channels such as upper LTP and lower LTP. The upper LTP are obtained by replacing the negative value into original LTP values with 0. The lower LTP are obtained in two step such as by replacing every value of 1 into the original LTP to get the value as 0, then the negative values are changed to 1.

From Hog and LTP, the normalized histogram-oriented features and texture features are obtained

from the LAB image. The features of HoG and LTP are fused into one matrix to get features vector by using Deep Convolutional Neural Network(D-CNN).

3.4 Feature fusion

After extracting the features, the obtained features from the proposed HoG and LTP are fused by using the D-CNN. The fusion process is adopted to extract the advantage of several patterns of descriptors that increases the accuracy.

3.4.1. Deep convolutional neural network (D-CNN)

CNN provides more attention to the identification of objects due to the ability to automatically finding certain features in the different categorizations of images. CNN includes 4 types of layers such as, the input images are passed and computed by the neurons by convolutional layers that are connected with the inputs' local regions. The neurons are calculated based on dot product among lesser weights and region which is connected with input volume. The activation is undergone by utilizing ReLU layers and it will not change the dimensions of input images. Pooling layers are performed to decrease the effect of noise among the extracted features. Finally, the higher levels of features are determined by utilizing the Fully Connected (FC) layer. The VGG19 and AlexNet are utilized in the proposed HoG and LTP with D-CNN for feature extraction. The AlexNet deep CNN consists of 5 convolutional layers, 3 pooling layers and 3 FC layers with softmax classification functions. The VGG19 network includes 16 convolutional layers, 3 FC layers and 19 learnable weight layers with the softmax function. The dropout regularization will be utilized by VGG19 in the FC layer and utilizes ReLU activation functions in the convolutional layer. These two models include convolutional, pooling, ReLU, normalization, and FC layer. The convolutional layer extracts the local features from images as shown in Eq. (17).

$$g_i^S = b_i^S + \sum_{j=1}^{m_1(S-1)} \psi_{i,j}^S \times h_j^{S-1} \tag{17}$$

Where, g_i^S is the S output layer, b_i^S is the base value,$\psi_{i,j}^S$ is the filter connected with j^{th} feature map and h_j is the $S-1$ output layer. The pooling layer extracts maximum responses from the lesser convolutional layer to reduce unwanted features and it solves the problem of overfitting which is explained through Eqs. (18-20).

$$m_1^S = m_1^{S-1} \tag{18}$$

$$m_2^S = \frac{m_2^{S-1} - F(S)}{S^L} + 1 \tag{19}$$

$$m_3^L = \frac{m_3^{S-1} - F(S)}{T^S} + 1 \tag{20}$$

Where, T^S are the strides, m_1^S, m_2^S, and m_3^S are the filter of feature maps, the other layers like ReLU and FC are explained in Eqs. (20, 21).

$$Re_i^l = max\left(h, h_i^{l-1}\right) \tag{21}$$

$$L_i^S = q\left(z_i^l\right) with\ z_i^l = \sum_{j=1}^{m_1(l-1)} \sum_{r=1}^{m_2^{l-1}} \sum_{s=1}^{m_3^{l-1}} w_{i,j,r,s}^l \left(L_i^{l-1}\right)_{r,s} \tag{22}$$

Where, Re_i^l is the ReLU layer, h is the output layer, L_i^S is the FC layer which follows pooling and convolutional layer that performs activation to FC layer for deeper feature extractions. The D-CNN integrated the oriented features and texture features obtained using HOG and LTP from LAB images. The D-CNN obtained more prominent feature information from oriented and texture features, which increases the accuracy of, object detection. The feature fusion helps to know the features of images fully for a description of their internal information. The features fused by using D-CNN are forwarded to the Region-based Convolutional Neural Network for detecting multiple objects.

3.5 Classification

The feature vectors obtained from the process of fusion are classified using R-CNN to detect multiple objects. The R-CNN uses CNN with the technique of regional proposal where the regions will classify as a detected object. The aim of using particular regions in R-CNN is to reduce the space of search as compared with the sliding windows-based approach that reduces the execution time of detection. The R-CNN has the advantage of semantic region identification which enhances the accuracy of detection and overcomes slower execution time from the predecessor. The R-CNN adds a pooling layer to fine-tune the Region of Interest (RoI) that claims the lapse of different stages. The pooling layer decreases the execution time of CNN, has R-CNN that utilizes different regions. The R-CNN selects Region Proposal Network (RPN) for a faster approach to region identification. The RPN adopts features from the convolutional layers that identify the bound boxes for the object proposal and split the features with the classification model. The RPN are trained separately for the detection of objects and combined with the test phase. The utilization of

CNN layers for the region's classification and proposal decreases the time of execution during the real-time performance. The RPN has a two-stage of detector that includes three major parts such as a shared bottom convolutional layer, RPN and classifier developed for RoI.

Initially, the input images are considered as multiple scales of feature map that combines various levels of deeper features. The RPN provides the objects of candidates based on a feature map. The single scale value of the feature map is replaced using faster R-CNN with the multiple scales of feature maps and a single scale of an anchor with a certain size of pixels. The R-CNN are applied on multiple scales of feature maps such as $\{P2, P3, P4, P5\}$ with the different receptive fields. . The RoI pooling is utilized to obtain the features that represent regions and RoI classifier which identifies the labels category based on features. The training loss of the classifiers is composed as shown in Eq. (23).

$$L_{\mathrm{Det}} = L_{RPN} + L_{RoI} \tag{23}$$

Where, L_{Det} is the overall training loss, L_{RPN} is the training loss of RPN, L_{RoI} is the training loss of RoI. The training loss of RPN and ROI includes two loss terms, one for the classification that accurately predicts the probability and the other is regression loss on the coordinates of boxes for effective localization. The RPN is trained separately to detect the objects and combined later in a testing phase. The RPN utilizes the features from the convolutional layer to predict the bounding boxes for multi-object detection and shares those features with the classification network. The utilization of the convolutional layer in RPN reduced the execution time and improve the object detection accuracy.

4 Results and discussion

Object detection is to identify the position and types of regions in the images or videos that attract humans. Many object detection methods were developed to classifying or detecting objects. However, the process of the assignment was time-consuming and laborious when manually annotated with the objects. The present research performs effective analysis on the caltech101 dataset for multi-object detection. In this research, proposed a feature extraction method using HoG and LTP for multi-object detection. The D-CNN network includes 5 convolutional layers, 3 pooling layers, 3 fully connected layers and 19 learnable weight layers. The network utilizes dropout regularization in a fully connected layer and applies ReLU activation on the convolution layer and the size of input training images are considered as $224 \times 224 \times 3$. The proposed method is evaluated by the python3 in windows 10, i7 core processor, 16 GB RAM, and 6 GB 2080Ti NVDIA GTX edition GPU environment. The performance metrics considered to evaluate the proposed method, where the quantitative and comparative analysis of the proposed feature extraction method is explained in this section.

4.1 Performance metrics

The parameters considered to evaluate the proposed feature extraction method by using HoG and LTP for multi-object detection is explained below,

- **Accuracy:** Accuracy is defined as the measure utilized to determine the exactness of the model. The accuracy is explained in Eq. (23).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100 \tag{23}$$

- **Precision:** Precision is defined as the ratio of truly predicted positive observation to the overall predicted observation for positives. The precision is explained in Eq. (24).

$$Precision = \frac{TP}{TP+FP} \times 100 \tag{24}$$

- **Recall:** Recall is defined as the ratio of truly predicted as the fault-modules which are explained in Eq. (25).

$$Recall = \frac{TP}{TP+FN} \times 100 \tag{25}$$

- **F-Measure:** F-Measure is the calculation of accuracy test which is defined as the mean of precision and recall. The F-Measure is explained in Eq. (26).

$$F - Measure = \frac{2PR}{P+R} \times 100 \tag{26}$$

4.2 Quantitative analysis

The values obtained for the proposed feature extraction method by using HoG and LTP for multi-object detection are shown in table 1. Table 1 includes the evaluation results for multi-object detection in terms of accuracy, precision, recall, and f-measure. The plotted graph for evaluated values of the proposed method is shown in Fig. 2.

Table 1 shows the performance of the proposed

Table 1. Performance evaluation of the proposed HoG and LTP for multi-object detection in terms of accuracy, precision, f-measure, and recall

Metrics	HoG	LTP	Proposed HoG and LTP
Accuracy (%)	87.56	88.25	92.48
Precision (%)	65.27	78.45	71.32
F-Measure (%)	83.98	86.35	90.45
Recall (%)	69.56	72.38	75.24
Time (Sec)	92	98	69

HoG and LTP method for multi-object detection in terms of performance measures such as accuracy, precision, f-measure, and recall. The accuracy achieved by the proposed HoG and LTP to detect multiple objects is 92.48%. The proposed HoG and LTP achieved a precision of 71.32%, f-measure of 90.45% and recall of 75.24% respectively. The time consumed by the proposed HoG and LTP with D-CNN is 69 seconds for the detection of multiple objects from the Caltech 101 dataset. Whereas, the HoG showed an accuracy of 87.56%, precision of 65.27%, f-measure of 83.98%, recall of 69.56% and acquired 92 seconds to detect the objects. Similarly, the LTP showed an accuracy of 88.25%, precision of 78.45%, F-measure of 86.35%, recall of 72.38% and acquired time of 98 seconds. Therefore, the proposed HoG and LTP showed effective performance by extracting the oriented and texture features. The

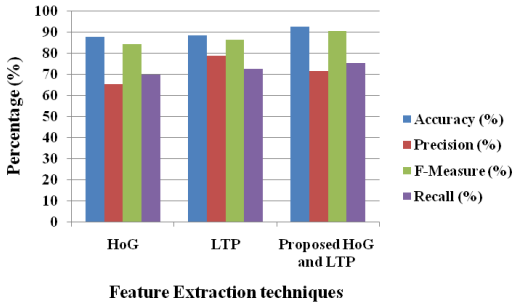

Figure. 3 The quantitative analysis result of the proposed HoG and LTP method

Table 2. Performance evaluation of the proposed HoG and LTP for feature fusion using R-CNN for multi-object detection in terms of accuracy, precision, f-measure and recall

Metrics	ANN	RNN	CNN	R-CNN
Accuracy (%)	85.65	86.64	90.34	92.48
Precision (%)	65.66	67.32	69.72	71.32
F-Measure (%)	83.18	84.45	88.32	90.45
Recall (%)	67.45	70.42	70.89	75.24
Time (Sec)	89	97	85	69

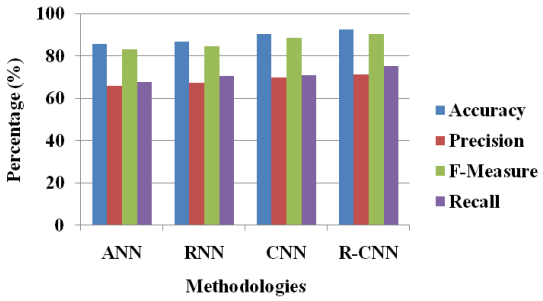

Figure. 4 The quantitative analysis of proposed feature extraction using HoG and LTP with R-CNN classification method

plotted graph for evaluated values of proposed HoG and LTP method is shown in Fig. 3.

Table 2. Shows the performance of the proposed HoG and LTP for feature fusion using R-CNN classification method for the detection of multi-object in terms of performance measures such as accuracy, precision, f-measure, recall and time. The Artificial Neural Network (ANN) showed an accuracy of 85.65%, precision of 65.66%, f-measure of 83.18%, recall of 67.45%, and acquired time of 89 seconds for multi-object detection. The Recurrent Neural Network (RNN) showed an accuracy of 86.64%, precision of 65.66%, f-measure of 83.18%, recall of 67.45% and consumed time of 97 seconds for multi-object detection. Similarly, the Convolutional Neural Network (CNN) showed an accuracy of 90.34%, precision of 69.72%, f-measure of 88.32%, recall of 70.89% and consumed time of 85 seconds. Whereas, the multi-object detection is performed by using the R-CNN classifier showed effective performance. The proposed HoG and LTP feature extraction techniques have the advantage of effectively extracting the oriented features and texture features from LAB images which are integrated to obtain prominent features thereby increasing the object detection performance. The graphical representation of quantitative analysis of the proposed HoG and LTP with R-CNN classifier is shown in Fig. 4.

4.3 Comparative analysis

The comparative analysis of the proposed HoG and LTP for multi-object detection is carried out and the values are tabulated as described in table 2. The existing techniques such as MODT [16] and D-CNN

Table 3. Comparative analysis of the proposed HoG and LTP with existing methods in terms of accuracy

Methods	Accuracy (%)
MODT[16]	76.23
D-CNN with SIFT[18]	89.7
Proposed HoG and LTP	92.48

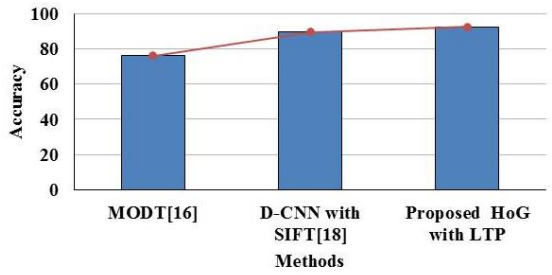

Figure. 5 The Comparative graph of the proposed method with existing methods in terms of accuracy

with SIFT features [18] are compared with the proposed multi-object detection approach.

Table 3 shows the comparison of proposed HoG and LTP for multi-object detection with the existing methods such as MODT [11] and D-CNN with SIFT features [13]. The rate of detection in the MODT method was low due to the failure of the tracking part. The D-CNN with LTP showed a problem in selecting the features because the size of inputs was different for every model that became a problem during fusion. The existing feature fusion method was unable to detect the similar appearances of objects and suffered from identifying the images in background which reduced the performance of object detection. The proposed HoG and LTP method extracted the oriented and texture features from images which are effective in detecting the of objects in images. The proposed HoG and LTP method effectively identify the similar appearance objects from the images. The proposed HoG and LTP method selected the features successfully and fused by using D-CNN method to detect the object. The proposed method achieved better accuracy of 92.48% for the detection of objects, whereas the existing MODT [11] and D-CNN with SIFT features [13] showed an accuracy of 76.23% and 89.7% respectively. Hence, the proposed HoG and LTP shows effective performance compared to existing methods. The plotted graph of accuracy values obtained for the proposed HoG and LTP with existing method is shown in Fig. 5.

5 Conclusion

The existing multiple object detection methods showed challenges due to degraded qualities of images such as blurriness and defocus which leads to unstable classification for similar objects. Further, the existing multi-object detection method showed lesser accuracy in detecting objects because it was not augmented the data properly due to large-scale variance and complex environments. To solve such issues, the proposed feature extraction method using HoG and LTP for multi-object detection. The Caltech101 dataset is used in the proposed research is converted to LAB and made the image look more vibrant to extract the features effectively. Then, the oriented and texture features are obtained by using the proposed HoG and LTP from pre-processed LAB images. Further, the features obtained from HoG and LTP are integrated into one matrix to get feature vectors by using D-CNN. Lastly, the R-CNN is used to classify the fused feature vectors obtained from the process of fusion to detect multiple objects. The proposed method effectively extracted the oriented and texture features that are integrated and obtained prominent features thereby increasing the object detection performance. The results showed that the proposed method achieved higher accuracy of 92.48% and precision of 71.32% in detecting multiple objects compared to existing methods.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

The paper background work, conceptualization, methodology, dataset collection, implementation, result analysis and comparison, preparing and editing draft, visualization have been done by first author. The supervision, review of work and project administration, have been done by second author.

References

[1] J. Wei, J. He, Y. Zhou, K. Chen, Z. Tang, and Z. Xiong, “Enhanced object detection with deep convolutional neural networks for advanced driving assistance”, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21, No. 4, pp. 1572-1583, 2019.

[2] P. Zhang, W. Liu, H. Lu, and C. Shen, “Salient object detection with lossless feature reflection and weighted structural loss”, *IEEE Transactions on Image Processing*, Vol. 28, No. 6, pp. 3048-3060, 2019.

[3] H. Lee, S. Eum, and H. Kwon, “Me r-cnn: Multi-expert r-cnn for object detection”, *IEEE Transactions on Image Processing*, Vol. 29, pp.1030-1044, 2019.

[4] J. H. Yoon, C. R. Lee, M. H. Yang, and K. J. Yoon, “Structural constraint data association for online multi-object tracking”, *International Journal of Computer Vision*, Vol. 127, No. 1, pp. 1-21, 2019.

[5] K. Yoon, D. Y. Kim, Y. C. Yoon, and M. Jeon, “Data association for multi-object tracking via deep neural networks”, *Sensors*, Vol. 19, No. 3, pp. 559, 2019.

[6] M. Sualeh, and G. W. Kim, "Dynamic multi-lidar based multiple object detection and tracking", *Sensors*, Vol. 19, No. 6, pp. 1474, 2019.

[7] P. Tang, C. Wang, X. Wang, W. Liu, W. Zeng, and J. Wang, "Object detection in videos by high quality object linking", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 42, No. 5, pp. 1272-1278, 2019.

[8] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation", In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580-587, 2014.

[9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, No. 6, pp. 1137-1149, 2016.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 37, No. 9, pp. 1904-1916, 2015.

[11] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks", *arXiv preprint arXiv:1605.06409*, 2016.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection", In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779-788, 2016.

[13] W. Fang, L. Wang, and P. Ren, "Tinier-YOLO: A real-time object detection method for constrained environments", *IEEE Access*, Vol. 8, pp. 1935-1944, 2019.

[14] B. Xue, and N. Tong, "DIOD: Fast and efficient weakly semi-supervised deep complex ISAR object detection", *IEEE Transactions on Cybernetics*, Vol. 49, No. 11, pp. 3991-4003, 2019.

[15] W. Tian, M. Lauer, and L. Chen, "Online multi-object tracking using joint domain information in traffic scenarios", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21, No. 1, pp. 374-384, 2019.

[16] M. Elhoseny, "Multi-object detection and tracking (MODT) machine learning model for real-time video surveillance systems", *Circuits, Systems, and Signal Processing*, Vol. 39, No. 2, pp. 611-630, 2020.

[17] K. Fu, Z. Chang, Y. Zhang, G. Xu, K. Zhang, and X. Sun, "Rotation-aware and multi-scale convolutional neural network for object detection in remote sensing images", *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 161, pp. 294-308, 2020.

[18] M. Rashid, M. A. Khan, M. Sharif, M. Raza, M. M. Sarfraz, and F. Afza, "Object detection and classification: a joint selection and fusion strategy of deep convolutional neural network and SIFT point features", *Multimedia Tools and Applications*, Vol. 78, No. 12, pp. 15751-15777, 2019.

[19] Z. Huang, J. Wang, X. Fu, T. Yu, Y. Guo, and R. Wang, "DC-SPP-YOLO: Dense connection and spatial pyramid pooling based YOLO for object detection", *Information Sciences*, Vol. 522, pp. 241-258, 2020.

[20] Z. Cai, and N. Vasconcelos, "Cascade R-CNN: high-quality object detection and instance segmentation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.