Applied Intelligence (2022) 52:10398–10416
https://doi.org/10.1007/s10489-021-03073-z

SIRA: Scale illumination rotation affine invariant mask R-CNN for pedestrian detection

Ujwalla Gawande[1] · Kamal Hajari[1] · Yogesh Golhar[2]

Accepted: 2 December 2021 / Published online: 13 January 2022

Abstract

In this paper, we resolve the challenging obstacle of detecting pedestrians with the ubiquity of irregularities in scale, rotation, and the illumination of the natural scene images natively. Pedestrian instances with such obstacles exhibit significantly unique characteristics. Thus, it strongly influences the performance of pedestrian detection techniques. We propose the new robust Scale Illumination Rotation and Affine invariant Mask R-CNN (SIRA M-RCNN) framework for overcoming the predecessor's difficulties. The first phase of the proposed system deals with illumination variation by histogram analysis. Further, we use the contourlet transformation, and the directional filter bank for the generation of the rotational invariant features. Finally, we use Affine Scale Invariant Feature Transform (ASIFT) to find points that are translation and scale-invariant. Extensive evaluation of the benchmark database will prove the effectiveness of SIRA M-RCNN. The experimental results achieve state-of-the-art performance and show a significant performance improvement in pedestrian detection.

Keywords Computer vision · Mask R-CNN · Pedestrian detection · Deep learning · CNN · Neural network

1 Introduction

In the past few decades, the world has been observing the threats to security in urban areas, which has expanded the pertinence of vision-based surveillance systems enough to detect pedestrians in high-density areas. Pedestrian detection has countless applications in computer vision. The obvious application is video surveillance [1].

Need and importance of Pedestrian detection Recently, due to the COVID-19 pandemic, surveillance systems have been used to monitor crowded places. It helps to identify hot spots (the areas where the virus infection spreads more opportunities). Owing to human interaction in highly congested areas, the chances of the spread of the COVID-19 virus increase. Therefore, the demand for pedestrian detection systems has increased. However, the current state-of-the-art installed system requires a conventional manual inspection of the video, which is in most cases time-consuming and infeasible. Pedestrian detection is an essential and important task in any intelligent video surveillance system because it provides information for semantic understanding of video scenes. The focus of the researchers is to make it smarter and use deep learning to move from passive surveillance to active surveillance. The author proposes several promising frameworks to improve the accuracy and speed of pedestrian detection. Nevertheless, the deep learning framework still has room for improvement.

Need and importance of Pedestrian detection The detection accuracy is affected by various changes such as human body appearance, trajectory, posture, abrupt motion, scale changes, complex background, pedestrian deformation, partial or complete occlusion, shadows, etc. In this area, it is still an unresolved problem. Different factors related to the video capture method, such as low frame rate, unavailability of color information, camera sensor stability, compression technology, etc., directly affect the quality of the video sequence. It may limit the design of object detection algorithms. In addition, different cameras have different sensors, lenses, resolutions, and frame rates, resulting in different image quality. A low-quality image sequence can affect moving object detection algorithms. In some cases, different parts of a moving object might have different movements in terms of speed and orientation. When detecting moving

✉ Ujwalla Gawande

[1] IT Department, YCCE, Maharashtra, India

[2] CSE Department, GHRIET, Maharashtra, India

objects in the presence of a moving camera, the need to estimate and compensate for camera movement is inevitable. However, it is not an easy task to do because of possible camera's depth changes and its complex movements Table 1.

The development history of CNN based deep learning architecture can be summarized as follows The state-of-the-art Convolutional Neural Network (CNN) architecture used for object detection is You Only Look Once (YOLO) [2]. Compared with other CNN-based object detection frameworks, it is the most accurate and fastest.

PP-YOLO or PaddlePaddle YOLO is the most efficient and fast object detection model in real-time [3]. YOLO has different variants, such as YOLO v1, YOLO v2, YOLO v3, YOLO v4, and YOLO v5 [4]. However, YOLO-based object detection fails in multiple object detection and occlusion. Specifically, Mask R-CNN performs better in crowded environments. We can also use it for semantic, instance segmentation, and classification [5, 6]. Other architectures are Faster R-CNN [7, 8], Fast R-CNN [9], Region-based Fully Convolutional Network (R-FCN) [10], Single Shot MultiBox Detector (SSD) [11], Fully Convolutional Network (FCN) [12], Deep Convolutional Generative Adversarial Network (DCGAN) [13], Residual Neural Network (ResNet) [14], GoogLeNet [15], Visual Geometry Group (VGG Net) [16], ZFNet [17], AlexNet [18], Deep Belief Network (DBN) [19], LeNet [20], etc. used for the pedestrian detection. These frameworks are significantly better than other neural network architectures, such as Support Vector Machine (SVM) [21], AdaBoost [22], Probabilistic Neural network (PNN) [23], Radial basis Neural Network (RBN) [24], Artificial neural network (ANN) [25].

The motivation for the proposed research work The Fig. 1 illustrates the motivation for the proposed contribution. Changes in the appearance of pedestrian instances make pedestrian detection difficult. In many cases, images of natural scenes usually vary in proportion, orientation, and lighting. These artifacts make pedestrian incident detection and classification a challenging task. Several other problems include 1) uneven lighting, 2) blurry and hazy appearance, 3) highlighting pedestrians, 4) changes in size with different posture changes, etc. Fig. 2 describes the key ideas of a new SIRA R-CNN framework, which is built on the Faster R-CNN pipeline [12]. The input image is first verified by the illumination recognition algorithm whether the input image needs illumination enhancement. The input image has low contrast, and then the histogram enhancement is performed on the input image. Afterwards, for the object proposal generation, first passes the process image through the bottom shared convolutional layers to extract its feature maps. Use these feature maps and scale rotation, affine invariant feature map for generation of to generate the final detection result, which is defined on the recommended size. We can always improve conclusive results through proposed feature maps and two-stage backbone networks, which are suitable for current input at certain scales. Therefore, SIRA R-CNN can achieve excellent detection performance in a wide range of input scales. Since SIRA R-CNN shares the convolutional

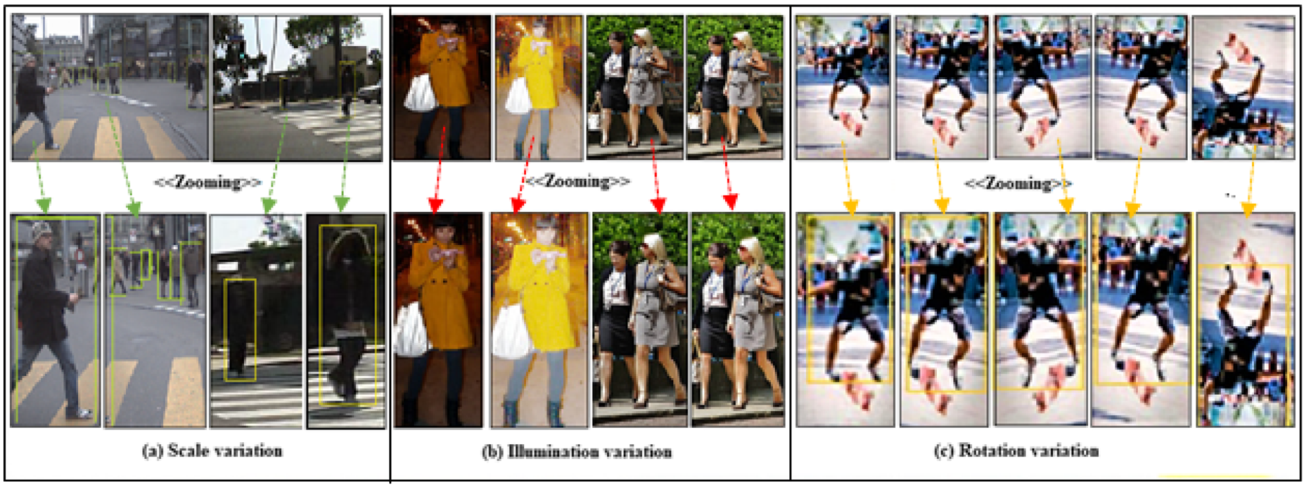


Fig. 1 Illustration of the motivation of the proposed SIRA Mask-RCNN. **(a) Scale variation:** The top row of the first column shows the pedestrian images from the California Institute of Technology [1] and the ETH [2] database. The first column and the bottom row show that the scaled visual appearance has changed significantly in proportion. **(b) Illumination variation:** The top row of the second column shows the pedestrian images from the California Institute of Technology [1] and the ETH [2] database. The bottom row of the second column shows a significant change in the visual appearance of the lighting after zooming. **(c) Rotation variation:** The top row of the third column shows the pedestrian images from the California Institute of Technology [1] and the ETH [2] database. The bottom row of the third column shows a significant change in the visual appearance when rotated after zooming

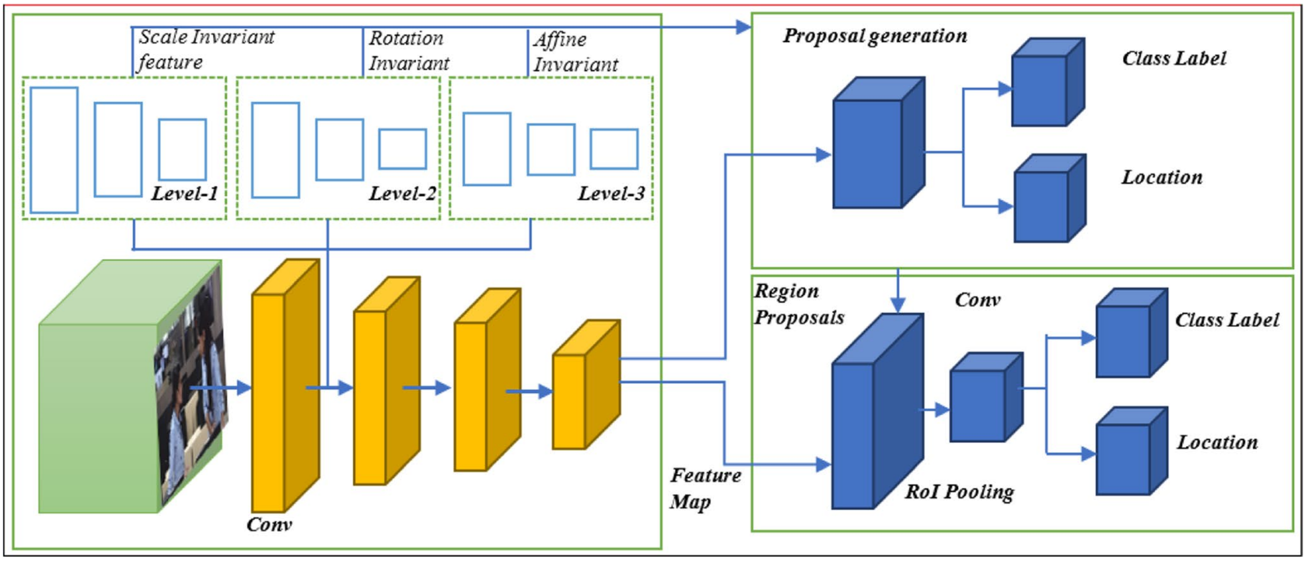


Fig. 2 Illustration of the proposed SIRA Mask R-CNN. We use multiple predefined box filters to filter multi-scale images to detect instances of different sizes. According to the proposed size of the target, the final result is obtained by fusing two-stage detectors and scale, illumination, rotation and affine invariant feature maps

Table 1 Feature extraction algorithm used in the state-of-the art pedestrian detector

Feature Extraction	Classifier	Reference
CNN	Mask R-CNN	[31]
HOG	Neural Network	[36]
LBP	SVM+CNN	[37]
Integral channel features	SVM+CNN	[38]
GLCM	SVM	[47]
Haar-wavelet transforms	SVM	[48]
Directional features	SVM	[46]
Vertical and horizontal	SVM	[47]
Edge intensities descriptors	Chamfer	[48]
Rectangular filter bank	Boosted cascade	[49]

features of the entire image with different target proposals, it is very effective in terms of training and testing time. Mask R-CNN has the advantage of predicting a binary mask for each specific detected object, while Faster-RCNN and FCN do not have this advantage. In these networks, objects are identified and classified, but no pixel-level segmentation mask is generated, and individual objects are not accurately distinguished.

The proposed region generation step generates an RoI, which may or may not contain the required objects. In the classification step, each RoI is classified as an object or background. However, despite its outstanding performance in terms of object detection accuracy, Mask RCNN is computationally costly for the new image as input to the system. The proposal area generation process takes time to generate the RoI. Similarly, due to the unavailability of meaningful information about the object's scale and rotation-invariant features, the existing Mask R-CNN cannot effectively classify the smallest rotating object. These two drawbacks restrict the use of the Mask R-CNN in a real-world application for object detection in video surveillance.

The main objective of this research paper is to propose a method to reduce the computational cost of the pre-trained Mask-RCNN in the testing phase and to detect the small, revolved object efficiently. The key idea of this method is to modify the version of the Region Proposal Network (RPN) to generate a feature map with a scale rotation affine invariant feature map, which will provide additional object scale, orientation information. In particular, the feature map of the network can locate objects in the image. Hence, proposal regions for a new input image generated with scale and orientation aware information in the trained network, which results in the efficient small and rotated object detection in less computational cost. The detection accuracy increases because of the availability of object information on multiple scales. Joint feature maps of multiple resolutions extracted from different layers of the two networks and used for pedestrian detection, which results in a low false-positive rate. Experimental results show improvements in detection rates due to the proposed feature map.

R. Girshick et al. introduced a Fast R-CNN [9] to resolve the problem of the scale-variance issue by applying brute-force data on the image for scaling at the cost of time and computational complexity. Y. Gong et al. [29] recommended to use a multi-scale filtering model for all objects of different sizes. However, Mask R-CNN is very popular in object detection. In this case, the recognition and generation of the object region proposal takes time to generate the ROI. Similarly, since there is no scaling factor, it will cause false detection of objects whose size changes during the object detection process. Each region proposal is convolved with a fixed-size filter mask to obtain CNN features. These features are inaccurate when the image has significant contrast, lighting, scale changes, and unclearness. These three shortcomings limit the application of Mask R-CNN in the real world.

The contributions of this paper are as follows:

1. A new SIRA Mask-RCNN is proposed to overcome scale, rotation and illumination artifacts.
2. In the proposed method. First, the illumination recognition and detection algorithms solve the problem of illumination changes. Secondly, the Contourlet transform and the direction filter bank use to solve the problem of rotation changes. Third, use ASIFT to solve the problem of translation and scale invariance.
3. The comprehensive evaluation of the proposed method on Caltech, INRIA, MS COCO, ETH, KITTI and the proposed academic environment database proves the effectiveness and robustness of the proposed SIRA Mask R-CNN.

The rest of the research paper is further organized as follows The latest architectural innovations in pedestrian detection technology and deep learning are described in Section 2. In Section 3, we describe the theoretical background of using deep neural networks for pedestrian detection. Section 4 describes the proposed approach to pedestrian detection. Section 5 presents the experimental results and comparative analysis of the proposed approach and the relevant methods. The last part ends with further research on the direction of pedestrian detection.

2 Literature survey

Researchers have proposed many pedestrian detection frameworks based on deep learning to improve the accuracy of pedestrian detection [30–35]. However, the accuracy of pedestrian detectors is affected by complex backgrounds, pedestrian scale changes, object occlusion, and illumination changes. These issues are partially addressed. The most commonly used method of pedestrian detection is the Histogram of Oriented Gradients (HOG) [36]. In this method, the directional gradient is used to detect objects in the image. It fails to detect an object in the case of occlusion and the HOG feature extraction process is time-consuming. Hence, it is not a perfect match for a real-time. In addition, In [37] proposed a combination method of HOG and local binary mode (LBP) to deal with partial occlusion of pedestrians. In this method, LBP is used to obtain the texture features of the image. The HOG features and texture features used together to detect the pedestrian. P. Dollar et al. [38] introduce the method of combining HOG features and color features. Compared with HOG features, this method is faster because it uses integral images. In this method, the feature pyramid based on directional gradients of different scales. This leads to fast feature calculations in real time. The texture and orientated gradient feature somewhat address the issues and challenges of pedestrian detection. Therefore, researchers continue to use region-based deep learning techniques to solve the problem of detectors based on texture and directional gradient features [39, 40]. However, the region-based techniques use the pedestrian region proposals. The results of these methods are more accurate and simpler. In [41], an instance segmentation using the coverage loss method for object segmentation was proposed.

In this method, CNN features and scale-invariant feature transform (SIFT) combined together to detect objects. The limitations of this method are 1) Due to the hierarchical structure, when objects in the scene are occluded, non-adjacent areas will not merge to form a single part. A model was proposed in [42], which integrates a patch-based CNN and a global multi-receptive field (MRF) network. In this method, patches of different sizes are used to identify the ROI. CNN is used to mark each patch. MRF contains the unidirectional graph of label patches. In this graph, each vertex depicts an instance label of each pixel. The limitations of this method include: 1) It is not suitable for detecting multiple related occluded objects. In [43] exhibited a Relief R2-CNN for pedestrian detection in real-time. In this method, the main aim was on faster ROIs identification using CNN features. The weaknesses of this approach are 1) not validated and tested in real-time. 2) the classification needs more time. In [31] present, a unified joint detection model for the cyclist pedestrian detection. This approach utilizes Fast R-CNN. The limitations of this method are 1) It is not verified in real time. 2) target detection rate of cyclic pedestrians could be improved. It needs the verification of the target detection method in intelligent driven vehicles. Farther, In [30] introduced employed Mask R-CNN and an Optical flow-based method for the detection of the active football player's pedestrian. The shortcoming of this method are 1) non-active football players have been misclassified as an active player. 2) segmentation process is time-consuming and manual. Hence, not suitable for real-time. In [30], a scale-aware Fast R-CNN (SAF R-CNN) method for pedestrian detection is introduced. In this approach, two sub-networks are employed to detect varying scale pedestrians. The major disadvantage of this method is that the training and testing time increase because of the utilization of a two-sub network and need time for computation. The proposed SIRA Mask R-CNN framework addresses the aforementioned problem by detecting the pedestrian at a different scale, orientation, and illumination. The next section describes the proposed SIRA Mask-RCNN framework (see Section 4).

3 Theoretical background

The conventional pedestrian detection system reported in the literature was classified into a Region of Interest (ROI) detection, feature extraction, and candidate classification. ROI detection is most often based on stereo vision. The most commonly used hand crafted features are HOG [36], LBP [37], Integral channel features [38], Gray level co-occurrence matrix [47], CNN features [31], HAAR-Wavelet [48], and Oriented gradients [49]. The classifier trained using these features. The different classifiers such as SVM, Mask R-CNN, Neural network, and Boosted cascade, etc. are used for the classification. The sliding windows [30] scheme and the hand-crafted feature-based method proposed, partially address the issues of pedestrian detection (1).

Hence, the researcher moves towards the region-based deep learning approaches to overcome the existing hand-crafted feature based system issues [39, 40] as shown in Table 2. LeNet is the first CNN architecture introduced by LeCun et al. [22] in 1998. It only includes two convolutional layers, a pooling layer, and a backpropagation network for training. LeNet has been trained on the revised National Institute of Standards and Technology (MNIST) database, which contains 50,000 images divided into 10 categories. At that time, it was successfully used for object detection and handwritten signature detection commercially. Its error rate is 26.2%. Next, AlexNet introduced by Krizhevsky et al. [20] in 2012. It is very similar to LeNet. AlexNet uses ReLU activation and cross-entropy loss function.

AlexNet has more convolutional layers. It is trained on the ImageNet large database and contains more than 1 million images from 1,000 categories. AlexNet has higher object detection accuracy than LeNet and is used for different types of object detection purposes. The error rate is reduced to 15.4%. Zeiler et al. [19] A modified version of AlexNet was launched in 2013. ZFNet uses 7 x 7 filters in the first convolutional layer instead of 11 x 11 as used in AlexNet. It achieves an error rate of 11.2%. Liu et al. [21] In 2009, an unsupervised deep neural network architecture was introduced. DBN consists of several layers along with multiple feature detectors or hidden units. Simonyan et al. citeref19 introduced VGGNet in 2014, which reduces the error rate to 7.2%. It expanded the number of convolutional layers to 19 layers. It limited the filter size reduced by a factor of 16 and to 3 x 3. GoogLeNet and ResNet use similar architectural patterns. Vanhoucke et al. [17, 18] The Google Inception network with VGGNet was introduced in 2014. It achieves the error rate of 6.7% marginally better than the VGGNet at the cost of complex architecture design patterns compares to VGGNet. He et al. [16] in 2015, introduced ResNet. It achieves an error rate of 3.57%. ResNet contains 152 convolutional layers. ResNet uses forward and backward instead of the initial model passes of the backpropagation algorithm. The problem with these approaches is the selection of a region before convolution. Objects in the image may have different spatial positions and aspect ratios. Need to select a huge area to get the object of interest in the image. Hence, an algorithm like YOLO, R-CNN developed to solve the problem of selection of regions that constitute an object of interest. Ross Girshick et al. [28] proposed R-CNN, in which a selective search algorithm is used to find a region of interest (RoI) in an image, which is called a region proposal. The selective search algorithm extracts 2000 regions from the image. The problem with R-CNN is that 1) it required an enormous amount of time to train the 2000 region proposals in the image. 2) We cannot use it for real-time applications. Because it takes 47 seconds for testing the image. 3) The selective search algorithm is fixed and not adaptive, so it sometimes generates bad region suggestions for complex images.

Fast R-CNN solves the shortcomings of R-CNN. Ross Girshick et al. [14] again proposed the enhanced version of R-CNN i.e. Fast R-CNN. It is similar to R-CNN, but instead of inputting regional suggestions into CNN, it inputs this architecture into CNN to generate convolutional feature maps. In this process, it max-pooled the input feature map to generate a 7 x 7 square area of fixed size. This area is inputted into the fully connected Softmax layer to predict the category of the object. Results have been represented using the bounding box. Fast R-CNN is more efficient and faster than R-CNN because it does not require 2000 region proposal training to classify objects. Both the architecture usages the selective search to find region proposals using a

Table 2 Deep learning architecture used for pedestrian detection

Architecture	Dataset	Error rate (%)
LeNet [22]	MNIST	26.2%
AlexNet [20]	ImageNet	15.4%
ZFNet [19]	ImageNet	11.2%
VGGNet [18]	MNIST	7.2%
Google Inception network with VGGNet [16]	ImageNet	6.7%
ResNet [15]	ImageNet	3.57%
Fast R-CNN [14]	MS COCO	3.1%
Faster R-CNN [13]	MS COCO	2.75%
Mask R-CNN [12]	MS COCO	2.67%

selective search algorithm. This process is slow and time-consuming that affects the performance of the network.

Shaoqing Ren et al. [12, 13] introduced a Faster R-CNN to solve the Fast R-CNN speed problem by eliminating the selective search algorithm. In this architecture, the network itself learns the region proposals instead of using a selective search algorithm to identify region proposals. It proposes to use a separate network for the recognition region in the convolutional feature map. Hence, it is faster than the Fast R-CNN but not useful in the real-time applications for object detection. In all previous versions of the R-CNN series, regions are used to locate objects in the image. Joseph Redmon et al. [8] introduced a new architecture, i.e. YOLO. Here, a single convolutional layer network predicts the bounding boxes and class labels of these boxes. The input image composed of the S x S grid and each grid, m bounding boxes generated. For each of the bounding boxes, the convolutional network generates the class probability and bounding box location values for locating the object in the image. YOLO processes 45 frames per second, which is faster than other object detection algorithms. The problem with YOLO is that it cannot detect small objects in the image due to the limitation of the spatial plane coordinate position of the algorithm. The semantic segmentation algorithms classify the object at the pixel level. Recent architecture, i.e., Mask-RCNN, has used for instance segmentation. Recognizing each instance of a single object in the image and locating each instance pixel is called instance segmentation. However, instance segmentation is difficult because it requires accurate positioning and detection of all moving and non-moving objects in the image. Hence, it combines object detection and semantic segmentation techniques. Here, object detection, object classification, and the use of bounding boxes for representation. In semantic segmentation, each pixel is classified into a meaningful group without distinguishing object instances.

4 Proposed methodology

The proposed SIRA Mask-RCNN framework includes three stages: 1) input image/video enhancement, 2) illumination, rotation, scaling and translation invariant feature extraction, and 3) classification using Mask R-CNN. We describe each of these steps in brief as follows.

4.1 Input image/video enhancement

The input to the proposed system is the image or frame of the video. If it is a video, we transform it into a frame. We have conducted experiments on the publicly obtainable standard database and on our database in an academic environment. The main steps of enhancement include the use of histogram analysis for illumination recognition and detection. These steps are essential for enhancing the input image/video quality. We describe each of the steps in the next subsection.

4.2 Illumination identification and detection

The main aim of the proposed algorithm is to identify input image/frame required illumination enhancement or not based on the histogram analysis. First, we divide the input image into three categories as 1) uneven image, 2) glare image, and 3) uniform image based on the histogram and pixel-level analysis. The input image/frame converted from RGB to HSV color space. We need the conversion because our focus is on the intensity of the image. We have plotted the histogram of a value change to identify the type of image. After several experiments on the different images, we have obtained the optimal thresholds used for distinguishing the darker and brighter intensity. We use the percentais ge of low pixel and high pixel values to classify the image into one of the three categories.

We describe the sequence of steps for lighting recognition and detection below.

The uneven intensity image composed of light that unevenly distributed in different intensity ranges. Glare images include specific areas in the image that are overexposed by light. Image specific areas comprise brighter pixels. The ideal image does not require the enhancement, it comprises the uniformly distributed intensity. if the input image has a high variance, it is the case of irregular brightness. After histogram analysis, we observed that the image background is uniform, and low variance means that the input image does not need to be enhanced. In the image below, you can view the three small peaks that process three separate illuminated areas. The highest peak in the center results from making all the dark region black pixels to the mean value. After several experiments, we found that a standard deviation above 25 can be called uneven illumination cases.

We can also identify the bright pixels easily when there is a glare on the image. Except for objects, the glare image looks like a good image. There can be a possibility that the image comprises high brightness regions, so if the mean of the input image is above 200, then it is true that there is no need to detect the object. We can eliminate those frames, as shown in Fig. 2.

Finally, by using the proposed illumination recognition and detection algorithm, it is detected whether illumination enhancement is needed. In the next section, we describe image enhancement using logarithmic transformation.

Algorithm 1: Illumination identification and detection

```
Input: $f(x, y)$: Read the input image or video
Output: $isEnhanReq$: Input Image require
        enhancement or not
Required parameter: $T1, T2, T3, T4, T5, T6, T7$:
                    Threshold for comparing
                    pixel intensity value.
Begin:
Read the input image or video.
  if $readFile$ is video then
    $f(x, y) \leftarrow \sum_{n=1}^{m} vf(x, y)$ Read all the frames.
else
    $f(x, y) \leftarrow If(x, y)$ Read input image.
end
Convert input image/frame into HSV space.
  $Hf(x, y) \leftarrow rgb2hsv(f(x, y))$
  Read each pixel location by row and column index
  position.
  for $i \leftarrow 1$ to $m$ do
    for $j \leftarrow 1$ to $n$ do
      Compare intensity value against threshold.
        if $f(xi, yj) \geq T1$ then
          Image having high pixel intensity value.
            $HPf(xi, yj) \leftarrow f(xi, yj)$
      else
          Image having low pixel intensity value.
            $LPf(xi, yj) \leftarrow f(xi, yj)$
      end
      $j++$ For reading next column pixel in
        image.
    end
    $i++$ For reading next row pixel in image.
end
Calculate percentage of High Pixel and Low Pixel.
  PerHP ← TotHPF / TotImgPixel * 100
  PerLP ← TotLPF / TotImgPixel * 100
  Reshape the image to uniform the input image
  resolution. Plot the histogram of image pixel
  intensity aganist count of intensity values.
  Check weather enhancement is required or not
  isEnhanReq ← False
  Check weather image having uneven illumination or
  not
  if $stDev \geq 25$ and $PerLP \geq T2$ and $PerHP \geq T3$
  then
    Set isEnhanReq ← True
end
Check weather image having glare
  if $mean\ f(x, y) \geq T4$ and $PerHP \geq T5$ then
    Set isEnhanReq ← True
end
Check weather image having uniform illumination
  if $stDev \leq 25$ and $PerLP \leq T6$ and $PerHP \leq T7$
  then
    Set isEnhanReq ← True
end
```

4.3 Image enhancement

Logarithmic transformation replaces all pixel values with their logarithmic values. We used it for image enhancement as it unfolds, dark pixels as compared to the bright pixel. We compute the log transformation using the following Eq. 1.

$$L(x, y) = s * \log(1 + F(x, y)) \tag{1}$$

Here, $F(x, y)$, input image and s is scaling constant and $L(x, y)$, output image. The value of s is determined such that, we get the maximum output corresponding to the bit size used. So, the s is calculated using the following Eq. 2.

$$s = 255/\log(1 + \max(F(x, y))) \tag{2}$$

However, for the computation of constant value s any pixel intensity is 0 then its corresponding log value will be infinite. Hence, we are adding 1 to each pixel intensity value at the time of log transformation that results in any pixel intensity value is 0, then it will become 1 and its log value will be 0. Fig. 3 shows the result input image enhancement after log transformation.

In the conversion, the darker low intensity is given the brighter high value, so that the features existing in the darker or gray area of the image are more visible and obvious to the human eye. It can also reduce the brighter brightness to a lower dark value. Results in a more illuminated image in the output as shown in Fig. 4.

However, the luminous intensity not scaled down to the low-intensity region. The scaling constant represents and transforms the higher intensities. In the next subsection, we have described the rotation invariant feature extraction.

4.4 Rotation invariant feature extraction

The rotation-invariant features extracted using contourlet transformation. It converts the enhanced image into a multi-resolution part. It is a combination of the Laplacian pyramid and direction filter bank. The features obtained from the contourlet transform are scale invariants. It can provide smooth edges in any direction. We have not implemented wavelet transformation, because we see edges, but the directional information about the edges unknown. It gives edges in four directions (vertical, horizontal, +45°, -45°). We extracted the contourlet transformation edges in eight directions. In Laplace decomposition, the two main scaling operations are to perform reduction and growth. Each of the operation describe in brief as follows.

1. Reduced operation employs a low-pass filter to the transformed image and down-samples by two factors.

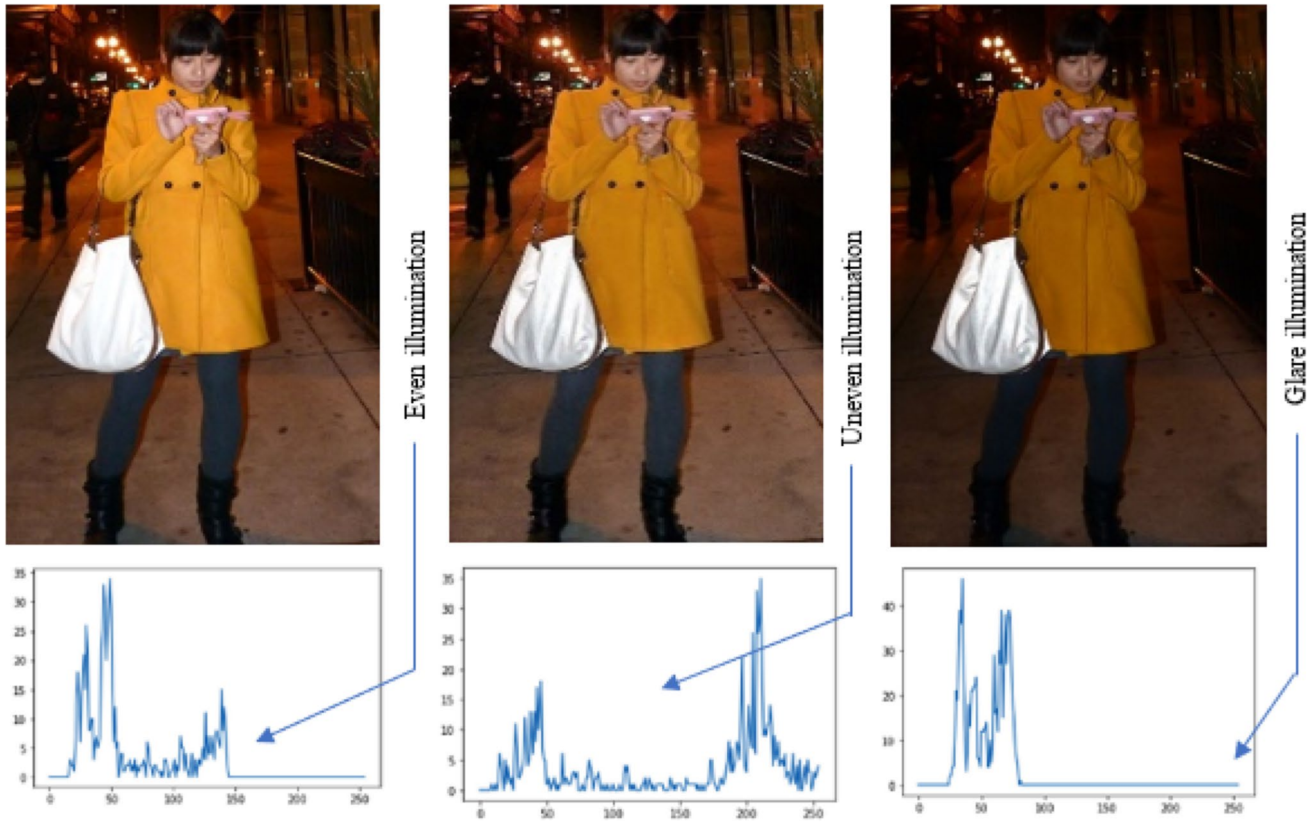


Fig. 3 First row shows even, uneven, and glare illuminated image. Second row shows the histogram of image

2. The grow operation applies a predefined interpolation technique and up-samples it by two factors.
3. Next, in the Gaussian pyramid a series of replica represented by $Gp_1, Gp_2, ..Gp_n$ of the transformed image represented by $lf(x, y)$ at different scale. It is accomplished by placing $Gp_1 = lf(x, y)$, and iteratively employing $Gp_{i+1} = reduce(Gp_i)$.
4. The Gaussian pyramid is processed to obtained Laplacian pyramids represented by $Lp_1, Lp_2, Lp_3...Lp_n$. It is accomplished by placing $Lp_n = Gp_n$, and iteratively employing $Gp_i = Lp_i$ grow Gp_{i+1}.

Later, the Laplacian pyramid feed into the directional filter bank (DFB). The input image obtained from the Laplacian pyramid of Level 1 modulated. In modulation, the constituent frequency component shifted before re-sampling the image component. In re-sampling, we have considered the frequency re-sampling matrix 2 x 2. It is used to reshape the dimand shap component to different parallelogram. At the end, down-sampled the image. It reduces the number of grid point in the original image. In down-sampling the smoothing and sharpening performed. Later We subtracted the filtered responses to obtain the contours. The obtained edges are in 8 direction. Each of the directional filter bank steps are describe in brief as follows.

1. The first operation is a Gaussian low-pass filtering for smoothing the image. It calculates the average of pixel and replace with the original pixel repeatedly applies for every pixel with cutoff frequency D_0 as 85. It is represented by $Lo(\omega)$. The transfer function of a Gaussian lowpass filter is defined as:

$$Lo(\omega) = e^{-D^2(u,v)/2D_0^2} \tag{3}$$

2. The next operation is a high-pass. It produces the Gaussian high frequency component and reduces low frequency component with cutoff frequency D_0 is 15. It is represented by $Ho(\omega)$. The transfer function of a Gaussian highpass filter is defined as:

$$Ho(\omega) = 1 - e^{-D^2(u,v)/2D_0^2} \tag{4}$$

3. At the end both values are subtracted to obtained the contours as Eq. 3. It is represented by $Co(\omega)$.

$$Co(\omega) = Ho(\omega) - Lo(\omega) \tag{5}$$

4. The same process continuously applies up to third iteration.

5. The three levels of information are extracted from the transformation. It is eight directional information. The sub-band of 0-3 is taken as horizontal direction and sub-band of 4-7 taken as vertical direction. Later these features are combined with CNN features of Mask R-CNN to obtained the detection result.
6. The histogram equalization is applied over the transformed image to obtained the normalized image.

Next, sub-section describes the ASIFT feature extraction and architecture of improved Mask R-CNN as shown Fig. 5 used for pedestrian detection and classification.

4.5 ASIFT feature extraction and mask R-CNN

The new scale-invariant feature map generation algorithm is placed in the Regional Proposal Network (RPN) to generate multi-scale feature maps for pedestrian instances of different scales. We divide the proposed framework into three stages viz. 1) scale-invariant feature map generation. 2) region proposal generation. 3) Extract the detected pedestrian target from the image. Input to the Mask R-CNN is the transformed image, and it generates the object surrounded by a bounding box, class label, and pixel-level mask. We used Mask-RCNN for pixel-level mask generation, as shown in Fig. 5. We describe the algorithm for generation of scale and translation invariant CNN feature as follows:

```
Algorithm 2: ASIFT feature extraction
Input: Image $u$, tilt factor $\delta = \sqrt{2}$, tilt range from 1
       to $n$ where, $n = 5$, rotation sampling step
       factor $b = 72$
Output: ASIFT key points features computed by
        tilt and rotation values
Begin:
For each orientation compute angle. for $t \leftarrow 1$ to $\delta m$
 do
    No need to compute the rotation points when
      $t = 1$, it means no tilt
      if $t = 1$ then
        $\theta \leftarrow 0$
        Compute scale, rotation, and
        translation-invariant features on the original
        image
        $key(t, \theta) \leftarrow \mathrm{SIFT}(u)$
    else
        Rotation angle for $\theta \leftarrow 0$ to $Nb/t$ do
            $\mathrm{u_r} \leftarrow \mathrm{rotate}(u, \theta)$
            Rotate image with bi linear
            interpolation
            $\mathrm{u_f} \leftarrow \mathrm{GaussianFilter}(\mathrm{u_r})$
            $\mathrm{u_t} \leftarrow \mathrm{tilt}(\mathrm{u_f}, \mathrm{t})$
            $key(t, \theta) \leftarrow \mathrm{SIFT}(u, \mathrm{u_t})$

        end
    end
end
```

As shown in Fig. 4, enhanced image is transformed by assuming all possible orientation and translation changes in the position. The tilt represented by t computed by Eq. 4.

$$t = \frac{1}{\cos\theta} \tag{6}$$

The directional transformation t on the input image $u(x, y)$ change to $u(x, y) \rightarrow u(tx, y)$. The gaussian filtering is used for reducing aliasing error. ASIFT key points, rotation invariant features, and CNN features are combined to obtained the detection result (Fig. 6).

We divided the process of pedestrian detection into three stages: 1) Region proposal generation, 2) the multi-scale, rotation, and affine invariant feature extraction. 3) Bounding box and category prediction. We convolve the different scale masks with various size anchor box filters to obtain the confidence score without omitting the low-resolution layer, which was omitted in the existing Mask R-CNN RPN network. Later, this score integrated with the rotation, scale, and affine invariant features obtained in an earlier stage. The primary purpose is to use the proposed method to improve the accuracy of regional recommendations. In RPN, we convolve the enhanced image with bottom-up and top-down methods to extract features. From the enhanced image, ResNet is used to extract high-level features.

The previous layer convolved with a $3x3$ sliding window to generate the multiple feature map. Next, apply the $1x1$ convolution filter again to the class prediction and bounding box. For the $3x3$ mask, we used a sliding window because the mask applied from the center will result in accurate information. K anchor boxes of different sizes used at the time of convolution so that different shape objects detected. We generate the output as the $2k$ score of the class layer and the $4k$ score of the bounding box regression. The scale-invariant feature map is fused with other feature maps to effectively locate objects of different scales. Next, each region's proposal aligned using the ROI align. In the end, we represent all the detected objects with the bounding box and segmentation mask at a pixel in the original image. At last, human detected class IDs separated from all detected objects.

5 Implementation

We assessed the performance of the proposed SIRA Mask R-CNN on benchmark pedestrian databases such as Caltech [1], INRIA [2], MS COCO [26], ETH [27], and KITTI [28] and our pedestrian database. The experiments and proposed deep learning framework implemented on single NVIDIA

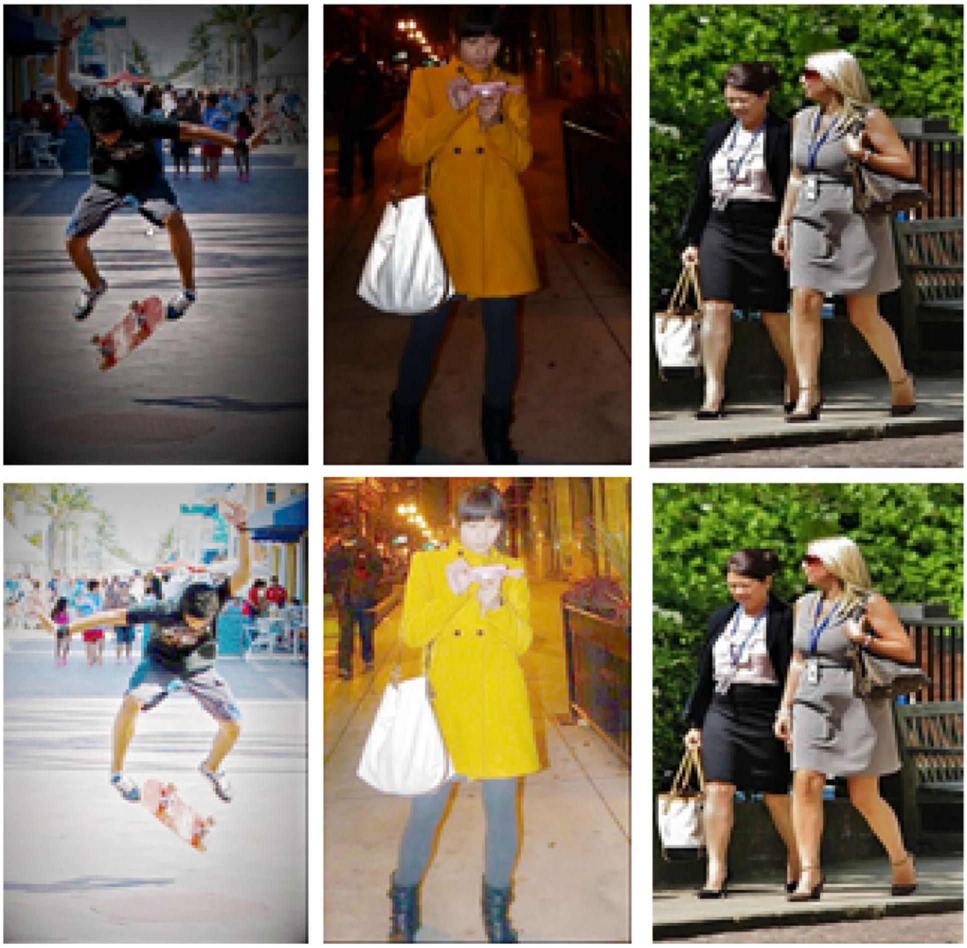


Fig. 4 Image enhancement results. First row shows before log transformation and Second row shows after log transformation

GPU and a CPU Intel Core i5 3.4GHz processor having 16GB RAM and 16GB NVIDIA graphics card.

6 Experiment

The State-of-the-art pedestrian analysis research conducted on publicly available benchmark data sets such as Caltech [1], INRIA [2], MS COCO [26], ETH [27], and KITTI [28], etc. The limitation of these datasets are: 1) the limited range of pedestrian poses recorded on the city streets in a controlled environment, 2) these datasets contain data with short lapses of time between successive observations of each ID in a single day, which allows to use clothing appearance features in identity matching, 3) All of these pedestrian data sets are recorded in various places such as streets and parking spaces, but they do not cover student behavior in an academic environment. This paper proposes a new dataset in an academic environment. Human experts annotated student pedestrian behavior on each frame sequence of the video, providing three types of information.

1. Pedestrian positioning with bounding box. The position of each pedestrian in the video frame is represented as a bounding box, and we can use this data for pedestrian detection, tracking, instance, and semantic segmentation.
2. Physical, behavioral, or adhered to human characteristics. Each pedestrian fully characterized by labels such as 'Face': eyes, eyebrows, forehead, nose, ears, mouth, facial hair, moustache, glasses, beard, 'hairstyle', 'hair color', 'age', 'body volume', 'gender', 'age', 'height', 'body accessories', 'ethnicity', 'head accessories', 'action' and 'clothing data'.
3. Annotated class label and ID. Each pedestrian has a unique identifier that is uniform across all video frames. These characteristics of the data set make it suitable for various recognition difficulties.

6.1 Experimental setup

We propose a pedestrian database composed of different behaviors of students under different conditions in academic activities such as students studying in a practical lab, examination hall scenarios, classrooms, a student cheating in an exam hall, a student taking an answer book outside the exam hall, a student stealing a mobile phone or other electronic

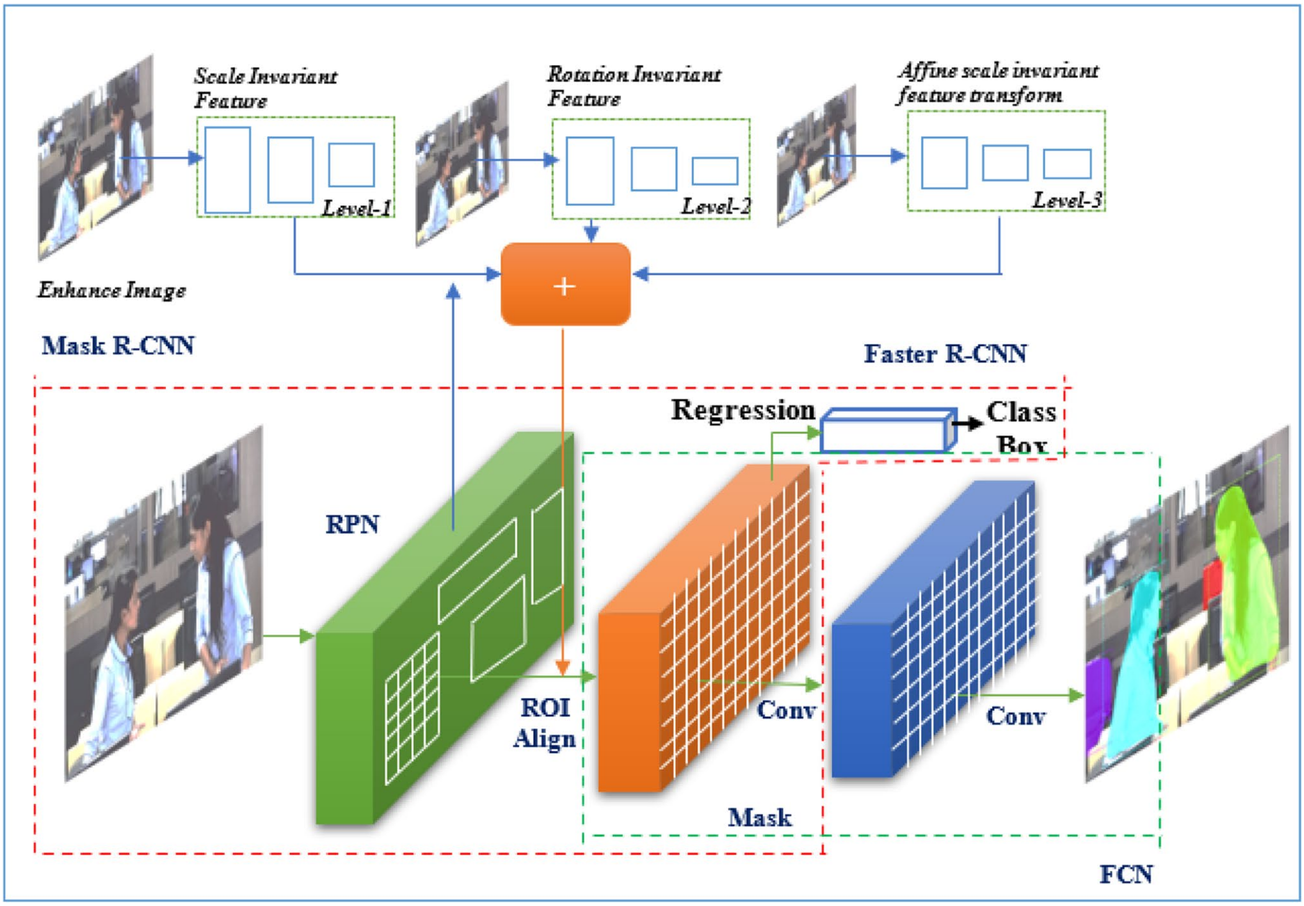
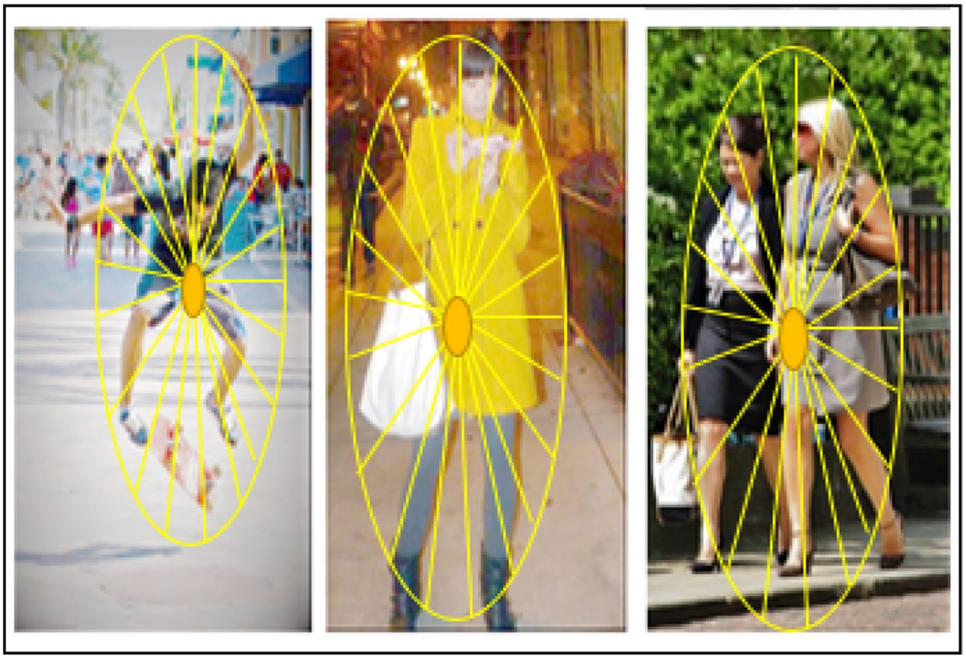


Fig. 5 SIRA Mask R-CNN: Improved Mask R-CNN Architecture

Fig. 6 Affine Scale Invariant Feature Map Generation. Point along the ray from initial point



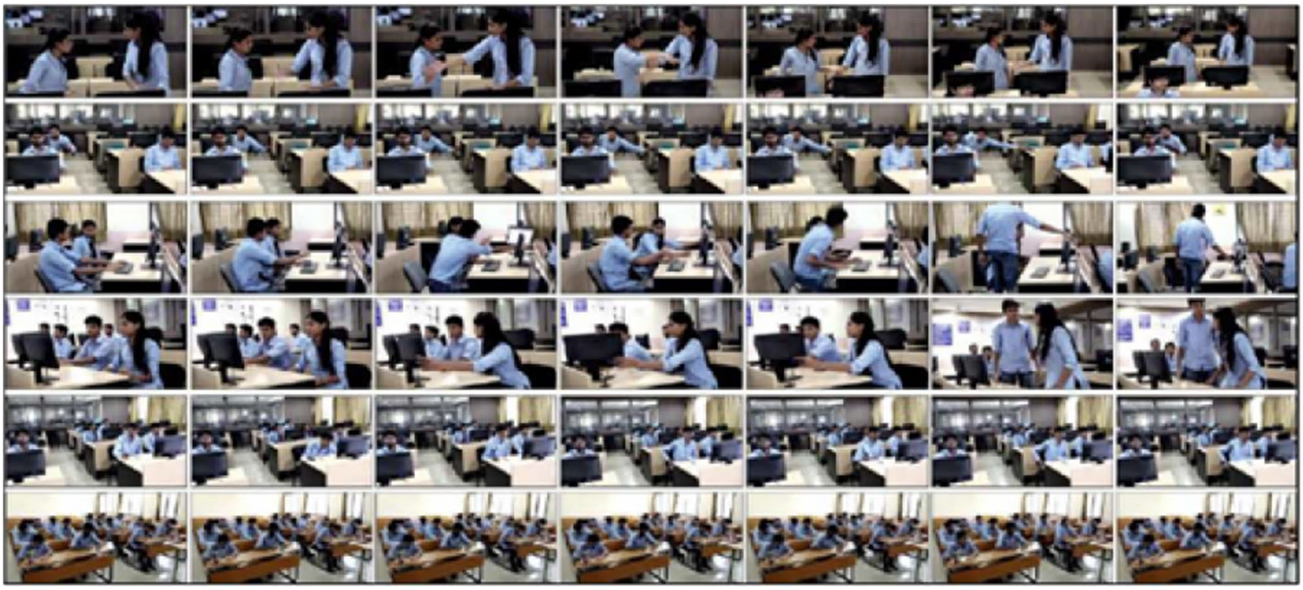


Fig. 7 Sample image of the proposed database. The first row illustrates two girls dispute in the lab. The second row illustrate the scenario stealing the mobile phone in lab. The third row illustrate a scenario of a student threatening. The fourth row shows the same threatening scenario(front view). The fifth row shows the scenario of students stealing the lab equipment. The sixth row shows the scenario cheating in the exam hall

devices such as a mouse or keyboard, a student stealing lab equipment, a student dispute on the college premises, a student disturbing another student, a student threatening another student, etc. as shown in Fig. 7. Students' behavior in college premises is recorded using a high-quality DSLR camera from different viewing angles. The video is recorded at 30 f/s. The database includes approximately 100 sample videos. 20 to 30 minute duration for each sample video.

6.1.1 Image/video acquisition framework

Student's behavior in college premises recorded using a high-quality DSLR camera from a different viewing angle. We recorded video at 30 f/s, enabled 4K recording, with a resolution of 3840×2160, and H.264 compressed to .mp4 format. The database includes more or less 100 sample videos. The duration of each example video is 20-30 minutes. The camera tilt angle varies from 45° to 90°. Pedestrians are students of Yeshwantrao Chavan College of Engineering in Nagpur, aged 22-27 years old, over 90%, of which 65% are male and 35% are female, mainly of Indian ethnicity. The key features of the data acquisition settings summarized in Table 3.

6.1.2 Data pre-processing

In the proposed dataset, we classified the video information namely into three directories as "Train", "Test" and "Validation". We describe the following three entities:

1. Annotations directory contains a XML file for each image. This file contains all the information about the image.
2. The frames are extracted from each video and divided into training and validation set in different directory.
3. The details about the frames are stored in the .txt file, it stores a unique identity number for each image.

However, for the classification purpose, we defined a SIRA Mask R-CNN model, along with "ReLU" as an activation function. In the proposed approach, we used sparse categorical cross-entropy instead of categorical cross-entropy for the compilation of SIRA Mask R-CNN model. The major advantage of using sparse classification cross-entropy is that it preserves time and computational effort, because it only uses a single value for a class label instead of the entire feature vector. We used simultaneous callbacks. A callback is a phenomenon that can execute processes at different stages of training phase. Each of these stages are describe as follows:

Table 3 The proposed dataset video Acquisition configuration

Parameter of Camera	Setting
Camera:DSLR HD:12.4M	Frame size:3840×2160
Lens FOV 94 20 mm f/2.8 focus	ISO Range: 100-3200
Camera tilt angle	45° to 90°
Video recording format	.mp4

Table 4 The proposed SIRA Mask R-CNN model configuration

Parameter	Setting
Backbone	ResNet101
Backbone Strides	[4,8,16,32,64]
Batch Size	1
BBox	[0.1 0.1 0.2 0.2]
Backbone shape	None
Max obj detection	100
Min Confidence	0.7
NMS Threshold	0.3
FPN Layer Size	1024
GPU	1
Gradient Norm	5.0
Image Per GPU	1.0
Image Channel Count	3
Image Max Dim	1024
Image Meta Size	93
Image Min Dim	800
Image Min Scale	0
Image Resize Mode	Square
Image Shape	[1024 1024 3]
Learning Momentum	0.9
Learning Rate	0.001
Loss Weight	rpnClassLoss: 1.0, rpnBboxLoss: 1.0, mrcnnClassLoss:1.0, mrccBboxLoss:1.0, mrcnnMaskLoss:1.0
Mask Pool Size	14
Mask Shape	[28 28]
Max GT Instances	100
Mean Pixel	[123.7 116.8 103.9]
Mini Mask Shape	[56 56]
Number of Class	2
Pool Size	7
Post NMS RoIs Inference	1000
Post NMS RoIs Training	2000
Pre NMS Limit	6000
RoI Positive Ratio	0.33
RPN Anchor Ratios	[0.5, 1, 2]
RPN Anchor Scales	[32, 64, 128, 256, 512]
RPN Anchor Stride	1
RPN BBox	[0.1 0.1 0.2 0.2]
RPN NMS Thresholds	0.7
RPN Train Anchors Per Image	256
RPN NMS Threshold	1000
Top Down Pyramid Size	256
Train BN	False
Train RoIs Per Image	200
USE Mini Mask	True
USE RPN RoIs	True
Validation Steps	50
Weight Decay	0.0001

1. Early stopping stage: When the observed indicators stop improving the results, the training process stops.
2. Reduce Learning rate On Plateau:When the observed indicator stops improving the result, it is used to reduce the learning rate.

The detailed configuration of the the proposed SIRA Mask R-CNN model illustrated in Table 4. We can see that most of the images classified by our model are correct. Also, as there is always room for improvement, this model can perform better with a larger dataset. The summary of layers with their required parameter while training and validation phase for the proposed SIRA Mask R-CNN shown in Table 5.

The model got a notable accuracy of 96% on the testing set and 95% of accuracy on the validation set. Next, we have computed the training loss and accuracy and validation loss accuracy. The Fig. 8(8) illustrate the accuracy and epochs plot for representing accuracy and validation accuracy and Fig. 8(8) illustrate the loss and epochs plot for representing loss and validation loss. The *loss* represented training loss. The *accuracy* represented training accuracy. The *valLoss* represented validation accuracy, and the *valAcc* represented validation accuracy.

Table 5 The proposed SIRA Mask R-CNN model summary

Layers type	Output Shape	Params
InputLayer	(None, 224, 224, 3)	0
Conv2D	(None, 224, 224, 64)	1792
Conv2D	(None, 224, 224, 64)	36928
MaxPooling2D	(None, 112, 112, 64)	0
Conv2D	(None, 112, 112, 128)	73856
Conv2D	(None, 112, 112, 128)	147584
MaxPooling2D	(None, 56, 56, 128)	0
Conv2D	(None, 56, 56, 256)	295168
Conv2D	(None, 56, 56, 256)	590080
Conv2D	(None, 56, 56, 256)	590080
MaxPooling2D	(None, 28, 28, 256)	0
Conv2D	(None, 28, 28, 512)	2359808
Conv2D	(None, 28, 28, 512)	2359808
Conv2D	(None, 14, 14, 512)	0
MaxPooling2D	(None, 14, 14, 512)	2359808
Conv2D	(None, 14, 14, 512)	2359808
Conv2D	(None, 14, 14, 512)	2359808
Conv2D	(None, 7, 7, 512)	0
MaxPooling2D	(None, 25088)	0
Flatten	(None, 4096)	102764544
Dense	(None, 4096)	16781312
Dense	(None, 2)	8194

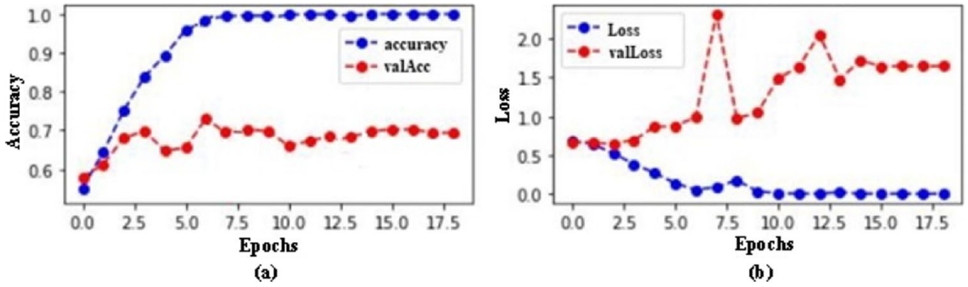


Fig. 8 Accuracy and loss training and validation. (a) Accuracy vs Epochs for representing accuracy and validation accuracy. (b) Loss vs Epochs for representing loss and validation loss

6.1.3 Pedestrian annotation

The proposed dataset completely annotated at the frame of video, by human specialists. We provide a csv file for each video sequence, using the same file naming protocol as videofilename.csv. The labeling process is divided into three stages: 1) human detection; 2) tracking, 3) suspicious activity recognition and soft biometric features. First, the Mask R-CNN [12] method is used to provide an initial estimate of the position of each pedestrian in the scene, and the data obtained are manually verified and corrected. Next, the deep sort method [14] provided the preparatory tracking information, which was again corrected manually. As a result of these two initial steps, we obtain a rectangular bounding box representing the region of interest (ROI) for each pedestrian in each frame. The final stage of the annotation process is carried out manually, where a human expert who personally knows the students of the college sets up ID information and characterizes the samples based on soft labels. Table 6 shows the detailed information of the label annotated for each pedestrian instance in the frame, as well as the ID information, the bounding box that defines the ROI, and the frame information. For each label, we also provide a list of its possible values.

6.2 Experimental results and analysis

AP (Average Accuracy) is a commonly used index to measure the accuracy of object detectors (such as Faster R-CNN, Fast R-CNN SSD, etc.). Average precision calculates the average precision value of the recall value in the range of 0 to 1. We considered the following criteria in our experiments.

1. IF IoU ≥ 0.5, classify the object detection as True Positive (TP). We have calculated AP at IoU .50 and .75. represented as AP_{50}, AP_{75}. Also, at across scales small, medium, and large, represented as AP_S, AP_M, AP_L as shown in Table 7.
2. If IoU < 0.5, then it is a wrong detection and classifies it as False Positive (FP).
3. When ground truth is present in the image and the model failed to detect the object, we classify it as False Negative (FN).

Table 6 The proposed dataset 16 annotated attribute with other soft biometric Labels of the pedestrian in the frame

Attributes	Values
Height	0→'Children', 1→'Short', 2→ 'Medium', 3→ 'Tall', 4→'Not known'.
Age	0→0-11, 1→12-17, 2→18-24, 3→25-34, 4→35-44, 5→45-54, 6→55-64, 7→greater than 65, 8→'Not known'.
Bounding Box	[x→Top Left; y→Top left row; h→Height; w→Width]
ID	1, 2, 3, 4, …. 'Not known'.
Feet	0→'Sport', 1→'Classic', 2→'High Heels', 3→'Boots', 4→'Sandals', 5→'Nothing', 6→'Not known'.
Frame	1, 2, 3, 4, ... 'n'.
Body Volume	0→'Thin', 1→'Medium', 2→'Fat', 3→'Not known'.
Hairstyle	0→'Bald', 1→'Short', 2→'Medium', 3→'Long', 4→'Horse Tail', 5→'Unknown'.
Hair Color	0→'Black', 1→'Brown', 2→'White', 3→'Red', 4→'Gray', 5→'Occluded', 6→'Not known'.
Head Accessories	0→'Hat', 1→'Scarf', 2→'Neckless', 3→'Occluded', 4→'Not known'.
Upper Body Clothing	0→'T-shirt', 1→'Blouse', 2→'Sweater', 3→'Coat', 4→'Bikini'. 5→'Naked', 6→'Dress', 7→'Uniform', 8→'Shirt', 9→'Suit', 10→'Hoodie', 11→'Cardigan'.
Lower Body Clothing	0→'Jeans', 1→'Leggins', 2→'Pants', 3→'Shorts', 4→'Skirt', 5→'Bikini' , 6→'Dress', 7→'Uniform', 8→'Suit', 9→'Not known'.
Moustache	0→'Yes', 1→'No', 2→'Not known'.
Beard	0→'Yes', 1→'No', 2→'Not known'.
Action	0→'Walk', 1→'Run', 2→'Standing', 3→'Sit', 4→'Cycle', 5→'Exercise', 6→'Pet', 7→'Phone', 8→'Leave Bag', 9→'Fall', 10→'Fight', 11→'Date', 12→'Offend', 13→'Trade'.
Accessories	0→'Bag', 1→'Backpack', 2→'Rolling', 3→'Umbrella', 4→'Sport', 5→'Market', 6→'Nothing', 7→'Unknown'.

Table 7 Evaluation metrics used for SIRA Mask R-CNN pedestrian detector

Metric	Annotation	Description
Avg. Precision	AP	AP at IoU=.50:.05:.95
$\text{AP}_{\text{IoU}=.50}$	AP_{50}	AP at IoU=.50
$\text{AP}_{\text{IoU}=.75}$	AP_{75}	AP at IoU=.75
AP_{Small}	AP_{S}	AP for small objects area $< 32^2$
$\text{AP}_{\text{Medium}}$	AP_{M}	AP for medium objects $< 32^2 <$ area $< 96^2$
AP_{Large}	AP_{L}	Area for large objects area $> 96^2$

4. True Negative (TN) is every part of the image where we did not predict an object. This metrics is not useful for object detection, hence we ignore TN. We have computed the precision, recall, and mean accuracy precision (mAP) using the mathematical model as:

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} \text{AP}_{\text{i}} \tag{9}$$

We compare SIRA Mask R-CNN to the state-of-the-art techniques in the area of instance segmentation in Table 8. Our proposed model outperform baseline variants of previous state-of-the-art models. This involves MNC [50] and FCIS [51], which are the winners of the COCO 2015 and 2016 challenges, respectively. SIRA Mask R-CNN with ResNet-101-FPN backbone is better than FCIS [51]. We achieve a benchmark AP of 79%, AP_{50} of 68%, and AP_{75} of 64%. Again, we validate our system at multiple scale and obtained the results as AP_{S} of 16.5%, AP_{M} of 39.5%, and AP_{L} of 54.5%. SIRA Mask R-CNN results are illustrated in Figure 16 under challenging scenarios.

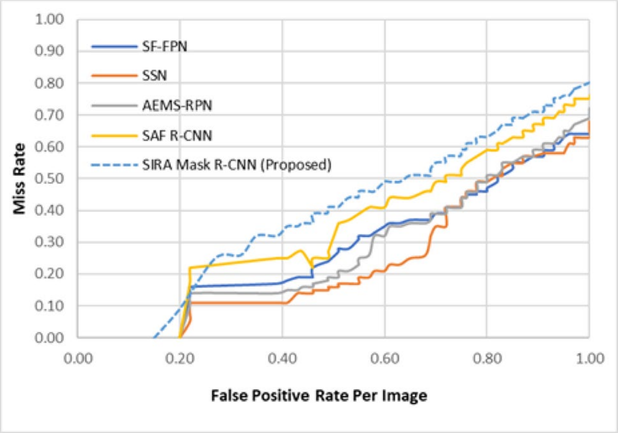


Fig. 9 Caltech dataset-SIRA Mask R-CNN gives the lowest log-average miss rate of 8.30%

6.3 Comparison with state-of-the-art pedestrian detection methods

6.3.1 Caltech

The proposed framework trained using Caltech database images. The Fig. 9 shows the proposed method is compared with the techniques such as SAF R-CNN [30], Side

Table 8 SIRA Mask R-CNN average precision results. Comparative analysis of average precision of proposed model to the state of the art approaches available in the literature

Methodology	Backbone	AP	AP_{50}	AP_{75}	AP_{S}	AP_{M}	AP_{L}
MNC [50]	ResNet-101-C4	24.6	44.3	24.8	4.7	25.9	43.6
FCIS [51]	ResNet-101-C5-dilated	29.2	49.5	-	7.1	31.3	50.0
Mask R-CNN [6]	ResNet-101-C4	33.1	54.9	34.8	12.1	35.6	51.1
Mask R-CNN [6]	ResNet-101-FPN	35.7	58.0	37.8	15.5	38.1	52.4
Mask R-CNN [6]	ResNeXt-101-FPN	37.1	60.0	39.4	16.9	39.9	53.5
R-CNN [4]	ResNeXt-101-FPN	61.61	50.13	44.79	–	–	–
pAUCEnsT [45]	ResNeXt-101-FPN	65.26	54.49	48.60	–	–	–
FilteredICF [49]	ResNeXt-101-FPN	67.65	56.75	51.12	–	–	–
DeepParts [4]	ResNeXt-101-FPN	70.49	58.67	52.78	–	–	–
CompACT-Deep [48]	ResNeXt-101-FPN	70.69	58.74	52.71	–	–	–
Regionlets [47]	ResNeXt-101-FPN	73.14	61.15	55.21	–	–	–
3DOP [46]	ResNeXt-101-FPN	77.93	65.01	60.42	–	–	–
SAF R-CNN [30]	ResNeXt-101-FPN	77.93	65.01	60.42	–	–	–
SIRA Mask R-CNN	ResNeXt-101-FPN	79	68	64	16.5	39.5	54.5

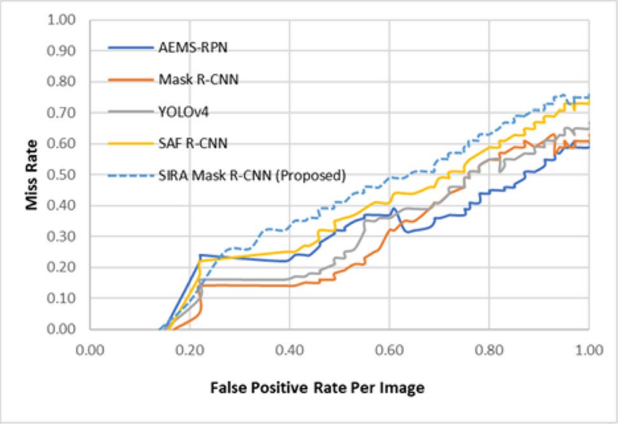


Fig. 10 INRIA dataset-SIRA Mask R-CNN gives the lowest log-average miss rate of 7.31%

Fusion FPN (SF-FPN) [44], Small-scale sense network (SSN) [45], Attention-Enhanced Multi-Scale Region, and Proposal Network (AEMS-RPN) [46]. It can be observed that SIRA Mask R-CNN outperforms the other methods by a huge margin. We achieve the lowest miss rate of 8.31%. It is the state-of-the-art performance for object detection by utilizing Mask R-CNN as shown in Table 9.

6.3.2 INRIA and ETH

The SIRA Mask R-CNN also trained and tested with the INRIA and ETH database images. The comparative results are shown in Figs. 10 and 11. First, for the INRIA dataset image, the proposed method gives the miss rate of 7.31%, which outperforms the existing method YOLOv4

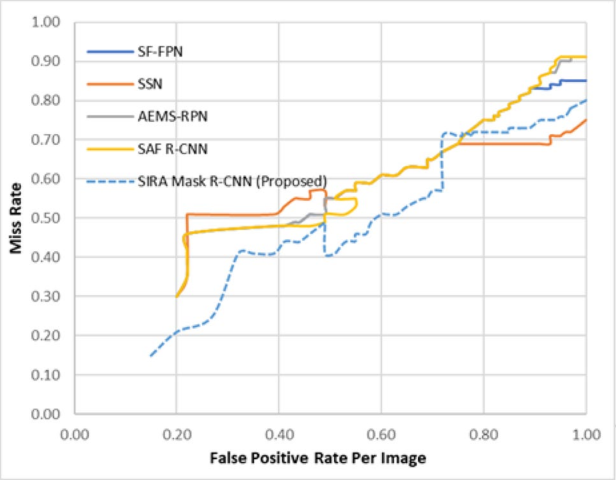


Fig. 11 ETH dataset-SIRA Mask R-CNN gives the miss rate of the proposed model is 32.63%

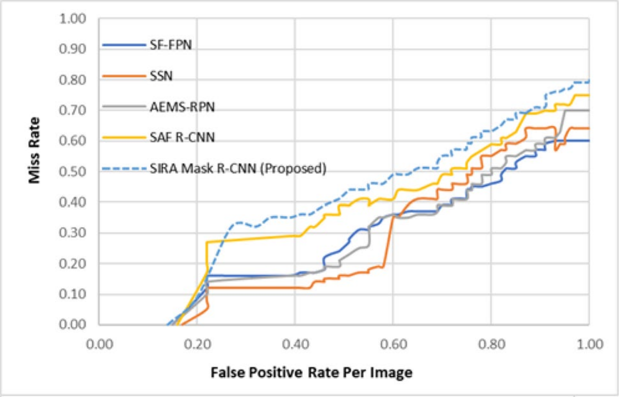


Fig. 12 KITTI dataset-SIRA Mask R-CNN gives promising results 79%, 68%, and 64%

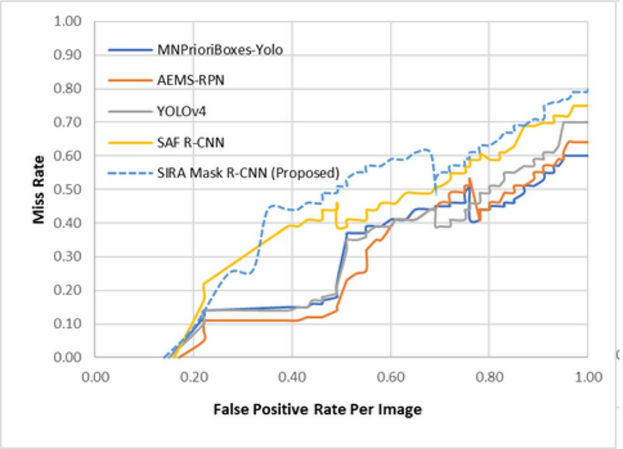


Fig. 13 MS COCO dataset-SIRA Mask R-CNN gives a miss rate of 8.56%

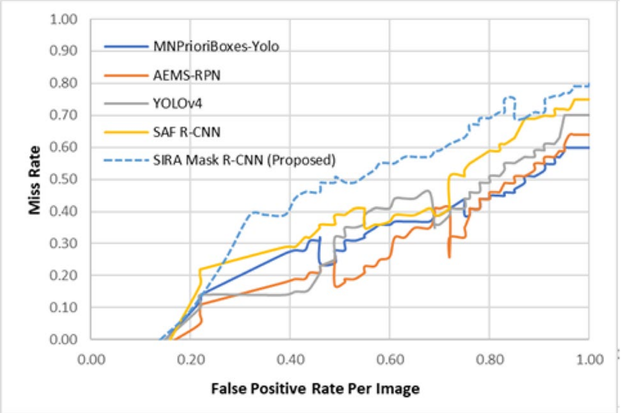


Fig. 14 Proposed pedestrian dataset-SIRA Mask R-CNN gives a miss rate of 8.68%

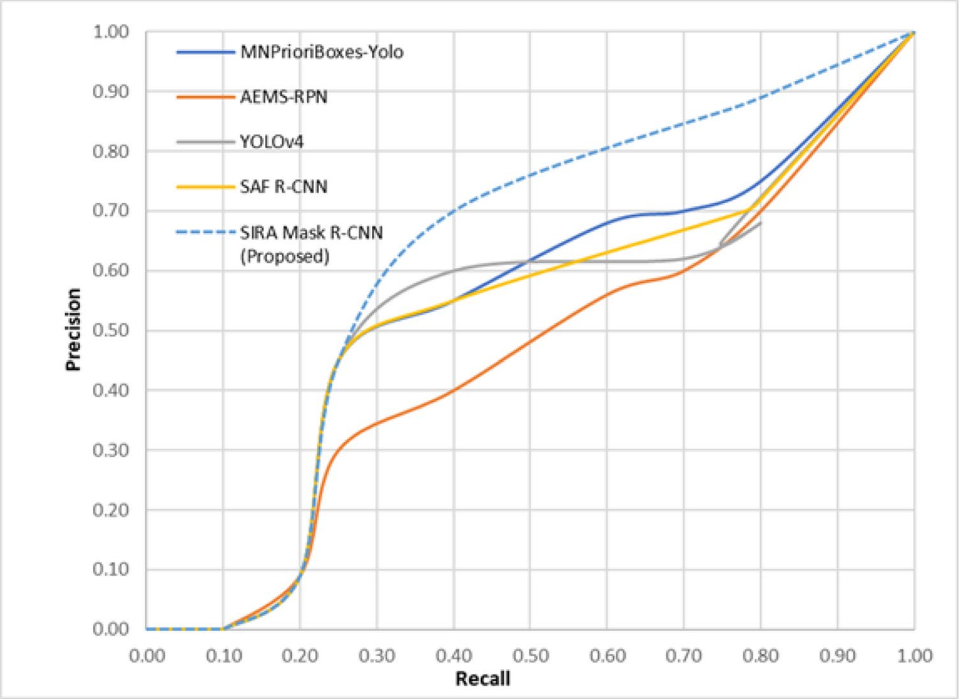


Fig. 15 Precision vs Recall: SIRA Mask R-CNN compare with recent state-of-the-art methods

[2], Mask R-CNN [5], SAF R-CNN [30], and AEMS-RPN [46]. Second, for the ETH dataset, the miss rate of the proposed model is 32.63% compared with 33.87% of [41] and 36.46% of [46]. In general, the proposed method achieves a higher detection rate on both the dataset and lower miss rate.

6.3.3 KITTI

The SIRA Mask R-CNN also tested on challenging the KITTI dataset. The pedestrian detection performance comparisons of the SIRA R-CNN with existing methods SAF R-CNN [30], SF-FPN [44], SSN [45], AEMS-RPN [46] are

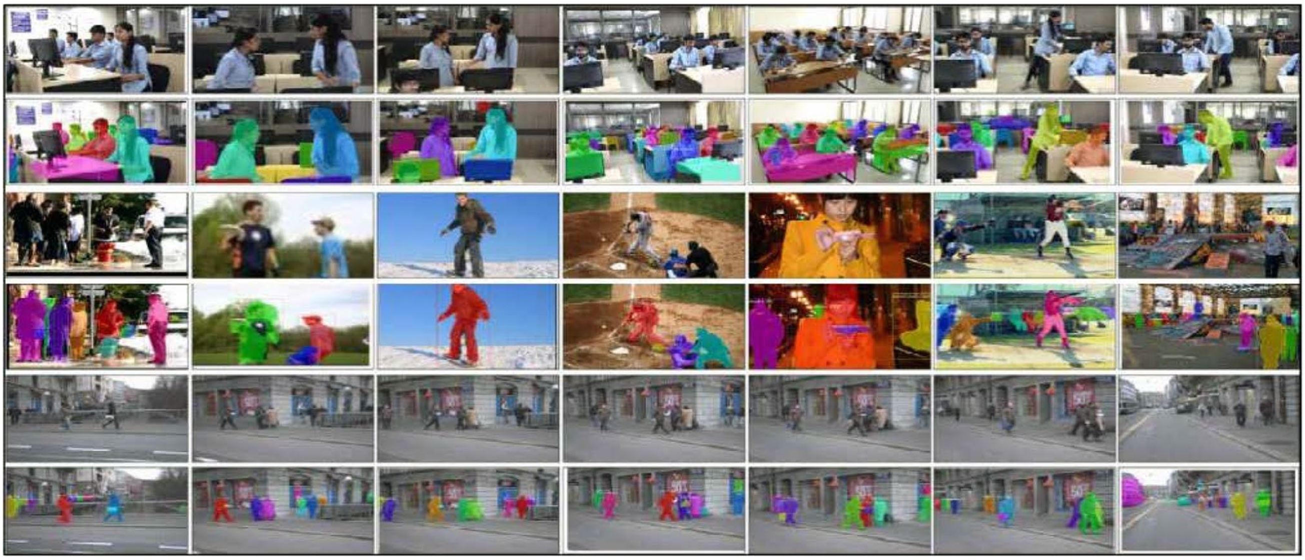


Fig. 16 SIRA Mask R-CNN result. First row - proposed database images. Second row-Results on proposed database. Third row-MS COCO dataset images [3]. Fourth row-Proposed method results on the Caltech dataset. Fifth row-Caltech dataset [1]. Sixth row-Proposed method results on the Caltech dataset

Table 9 SIRA Mask R-CNN pedestrian detector compare with the state-of-the art pedestrian detector

Framework, model	Miss rate(%)	Test Time(%)
Faster R-CNN [7]	17.60	0.22
Fast R-CNN Single scale [9]	13.70	0.34
YOLO [2]	11.67	3.04
R-CNN [9]	12.77	5.31
SAF R-CNN [31]	9.32	0.59
SIRA Mask R-CNN	8.31	0.50

shown in Fig. 12. The proposed approach gives promising results on the KITTI dataset, i.e., 79%, 68%, and 64%.

6.3.4 MS COCO and proposed pedestrian dataset

The results are shown in Figs. 13 and 14 The proposed approach compared with the existing methods in YOLOv4 [2], SAF R-CNN [30], MNPrioriBoxes-Yolo [41], and AEMS-RPN [46]. It gives a miss rate of 8.56% on MS COCO and miss rate of 8.68% on the proposed pedestrian dataset.

Figure 15 shows the results of the recall and precision of the proposed model compare with YOLOv4 [2], SAF R-CNN [30], MNPrioriBoxes-Yolo [41], and AEMS-RPN [46].

It can be observe that the SIRA Mask R-CNN outperform existing methods in terms of accuracy, speed, and time needed for pedestrian detection as shown in Fig. 15 on various pedestrian dataset. Table 9 illustrate the comparative analysis of proposed SIRA Mask RCNN with the state of the art deep learning framework such as Faster R-CNN [7], YOLO [2], Fast R-CNN [9], SAF R-CNN [30]. The proposed SIRA Mask R-CNN detect the pedestrian in presence of artifacts efficiently. The pedestrian detection accuracy is 96%, which is notable accuracy on challenging benchmark pedestrian databases.

7 Conclusion and future scope

In this paper, we proposed a scale, illumination, rotation, and affine invariant Mask-RCNN based pedestrian detector. The proposed framework detects the pedestrian in presence of scale, rotation, illumination artifacts efficiently.It delivers competitive results on the benchmark datasets such as Caltech [1], INRIA [2], MS COCO [26], ETH [27], KITTI [28], and our proposed academic environment database. The experimental results confirmed that the proposed SIRA Mask R-CNN delivers 1) the lowest miss rate of 8.31% on the Caltech dataset, 2) the lowest log-average miss rate of 7.31% INRIA, 3) miss rate of 32.63% on ETH dataset, the pedestrian detection accuracy of 79% on the KITTI dataset and 4) miss rate of 8.68% on the proposed database. The proposed method is superior in detecting the different sizes and varying illuminated pedestrian along with variation in orientation compared with the existing state-of-the-art techniques such as YOLOv4 [2], Mask R-CNN [5], SAF R-CNN [30], MNPrioriBoxes-Yolo [41], SSN [45], and AEMS-RPN [46].

In the future, the proposed framework can be enhanced from the following aspects. First, the proposed work addresses scaling, rotation, and lighting issues. Other problems, such as motion blur, partial or complete occlusion, can be effectively solved by extracting different features of pedestrian objects that are invariant to occlusion and motion blur to improve the feature extraction process. It should improve the pedestrian detection time to meet the real-time usage. An unsupervised approach to pedestrian detection can be useful to reduce the time of detection. The proposed model can also be used to detect human poses and trajectories of different sizes and to block pedestrians.

Compliance with ethical standards

Competing interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Dollar P, Wojek C, Schiele B, Perona P (2012) Pedestrian detection: An evaluation of the state of the art. TPAMI 34(4):743–761
2. Bochkovskiy A (2020) “YOLOv4: Optimal Speed and Accuracy of Object Detection”, CVPR, 23 April 2020
3. Long X (2020) “PP-YOLO: An Effective and Efficient Implementation of Object Detector”, CVPR, 3 August 2020
4. Redmon J, Divvala S, Girshick R, Farhadi A (2016) “YOLO”, CVPR, pp. 1-10, Las Vegas, Nevada, USA, 26 June-1 July 2016
5. He K, Gkioxari G, Dollar P, Girshick R (2020) Mask R-CNN. TPAMI 42(2):386–397
6. He K, Gkioxari G, Dollar and Girshick (2017) “Mask R-CNN”, ICCV, pp. 2980-2988, Italy, 22-29 October 2017
7. Ren S, He K, Girshick R, Sun J (2017) “Faster R-CNN’’. IEEE Trans Pattern Ana Machine Intell (TPAMI) 39(6):1137–1149
8. Ren S, He K, Girshick R, Sun J (2015) “Faster R-CNN”, NIPS, Montreal, pp. 1-9, Canada, 7-12 December 2015
9. Girshick R (2015)“Fast R-CNN”, ICCV, pp 1441-1448, Santiago, Chile 7-13 December 2015
10. Dai J, Li Y, He K, Sun J (2016) “R-FCN: Object Detection via Region-based FCN”, CVPR, pp 1-11, USA, July 2016
11. Liu, Anguelov D, Erhan D, Szegedy C (2016) “SSD”, ECCV, pp 1-17, Netherlands, 11-14 October 2016
12. Anguelov J, Shelh E, Darrell T (2015) “FCN for Semantic Segmentation”, CVPR, pp1-10, Boston, 8-10 June 2015
13. Radford A, Metz L, Chintala S (2015) Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, CVPR, pp 1-10, Boston, Massachusetts, 8-10 June 2015
14. He K, Zhang X, Ren S, Sun J (2015) Deep Residual Learning, CVPR, pp 1-10, Boston, 8-10 June 2015

15. Szegedy C, Ioffe S, Vanhoucke V, Alemi A (2016) Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning, CVPR, pp 1-10, LA, Nevada, USA, 26 June-1 July 2016
16. K S, Zisserman A (2015) Very Deep Convolutional Networks for Large-Scale Image Recog., CVPR, pp 1-10, Boston, 8-10 June 2015
17. Zeiler M, Fergus (2013) Visualizing and Understanding Convolutional Networks, CVPR, pp 1-11, USA, June 2013
18. Alex K, Sutskever I, Hinton G (2012) ImageNet Classification with Deep Convolutional Neural Networks, NIPS, pp 1-9, Nevada, United States, 3-6 December 2012
19. Kangming L (2016) Research on an improved pedestrian detection method based on DBN classification algorithm. RISTI 17(3):77–87
20. Lecun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11):2278–2324
21. Kang S, Byun H, Lee S (2002) Real-Time Pedestrian Detection Using Support Vector Machines, First International Workshop on SVM: Pattern Recog. with SVM, pp 268-277, Canada, 10 August 2002
22. Geronimo D, Sappa A, Lopez A, Ponsa D (2006) Pedestrian detection using AdaBoost learning of features and vehicle pitch estimation, International Conference on Visualization, Image and Image Processing, pp 1-8, Spain, 28-30 August 2006
23. Wu C, Yue J, Wang L, Lyu F (2019) Detection and Classification of Recessive Weakness in Superbuck Converter Based on WPD-PCA and Probabilistic Neural Network. MDPI Electronics 8(290):1–17
24. Alireza A, Mollaie, Reza, Yasser, Andi, Hosein (2011) Improved Object Tracking Using RBFNN, MVIP, Iran, pp 1-5, 16-17 November 2011
25. Emil N, Neghina, Mihai (2009) A NN approach to pedestrian detection, ICCOMP, pp 374-379, USA, 23 July 2009
26. Lin T, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Piotr Dollar, Lawrence Zitnick C (2014) MS COCO:, ECCV, Springer, Switzerland, pp 1-15, 6-12 September 2014
27. Ess A, Leibe B, Gool L (2017) Depth and appearance for mobile scene analysis, ICCV, pp 1-8, Italy, 22-29 October 2017
28. Geiger A, Lenz P, Urtasun R (2012) Are we ready for autonomous driving? the KITTI vision benchmark suite, CVPR, pp 3354-3361, RI, USA, 16-21 June 2012
29. Gong Y, Wang L, Guo R, Lazebnik S (2014) Multiscale orderless pooling of deep convolutional activation features, ECCV, pp 392-407, Zurich, Switzerland, 6-12 September 2014
30. Li J, Li X (2018) Scale-aware Fast R-CNN for Pedestrian Detec. IEEE Trans Multi 20(4):985–996
31. Wang K, Zhou W (2019) Pedestrian and cyclist detection based on deep neural network fast R-CNN. Inter. Journal of Advanced Robotic Systems, SAGE 16(2):1–10
32. Pobar M, Kosm M (2018) Mask R-CNN and Optical flow-based method for detection and marking of handball actions, CISP-BMEI 2018, pp 1-6, China, 13-15 October 2018
33. Minkesh A, Worranitta K, Taizo M (2019) Human extraction and scene transition utilizing Mask R-CNN, CVPR, pp 1-6, CA, United States, 16-20 June 2019
34. Gawande, Ujwalla, Hajari, Kamal, Golhar, Yogesh (2020) Pedestrian Detection and Tracking in Video Surveillance System: Issues, Comprehensive Review, and Challenges, Recent Trends in Computational Intelligence, IntechOpen, 1-24, April 2020
35. Gawande U, Hajari K, Golhar Y (2020) Deep Learning Approach to Key Frame Detection in Human Action Videos, Recent Trends in Computational Intelligence, IntechOpen, 1-17 February 2020
36. Dalal N, Triggs B (2005) HOG for Human Detection, CVPR, pp 886-893, CA, USA, 20-25 June 2005
37. Wang X, Han TX, Yan S (2009) An HOG-LBP human detector with partial occlusion handling, ICCV, pp 32-39, Japan, 29 September-2 October 2009
38. Dollar P, Tu Z, Per P, Bel S (2009) Integral channel features, BMVC, UK, pp 1-11, 7-10 September 2009
39. Gawande U, Golhar Y (2018) Biometric security system: a rigorous review of unimodal and multimodal biometrics techniques. IJBM, InderScience 10(2):142–175
40. Sil N, Son D (2014) Fergus. Instance Seg. of Indoor Scenes Using a Coverage Loss, ECCV, Switzerland
41. Li GY, Yifan Q, Xingda (2019) "Deep Learning Approaches on Pedestrian Detection in Hazy Weather", IEEE Transaction on Industrial Electronics, IEEE Transactions on Industrial Electronics, vol 1, 1 November 2019
42. Wang X, Wang M, Li W (2014) Scene-specific pedestrian detection for static video. TPAMI 36(2):361–374
43. Tian Y, Luo P, Wang X, Tang X (2015) Pedestrian detection aided by deep learning semantic tasks, CVPR, pp 1-10, Boston, 8-10 June 2015
44. Xu C, Wang G, Yan S, Yu J, Zhang B, Dai S, Li Y, Xu L (2020) "Fast Vehicle and Pedestrian Detection Using Improved Mask R-CNN", Mathematical Problems in Engineering, vol 2020, Hindawi, pp 1-15, 2020
45. Han B, Wang Y, Yang Z, Gao X (2020) "Small-Scale Pedestrian Detection Based on Deep Neural Network", IEEE Transactions on Intelligent Transportation Systems, 21:(7), 2020
46. Wang H, Wang LY, Shengjin (2019) "Fast Pedestrian Detection With Attention-Enhanced Multi-Scale RPN and Soft-Cascaded Decision Trees", IEEE Transactions on Intelligent Transportation Systems vol 1, No 8, 2019
47. Sharif M, Khan MA, Akram T et al (2017) A framework of human detection and action recognition based on uniform segmentation and combination of Euclidean distance and joint entropy-based features selection. J Image Video Proc. 2017:89
48. Wojek C, Walk S, Schiele B (2009) "Multi-cue onboard pedestrian detection," IEEE Conference on Computer Vision and Pattern Recognition, 2009
49. Zhao L, Thorpe C (2000) Stereo and neural network-based pedestrian detection. IEEE Trans Intell Trans Syst 1(3):148–154
50. Dai J, He K, Sun J (2016) "Instance-aware semantic segmentation via multi-task network cascades", In: Computer Vision and Pattern Recognition (CVPR), 2016
51. Li Y, Qi H, Dai J, Ji X, Wei Y (2017) "Fully convolutional instanceaware semantic segmentation", In: Computer Vision and Pattern Recognition (CVPR), 2017