Neural Computing and Applications (2023) 35:12729–12750
https://doi.org/10.1007/s00521-023-08390-7

ORIGINAL ARTICLE

Novel deeper AWRDNet: adverse weather-affected night scene restorator cum detector net for accurate object detection

Anu Singha[1,2] · Mrinal Kanti Bhowmik[1]

Received: 2 September 2022 / Accepted: 13 February 2023 / Published online: 7 March 2023

Abstract
Object detection in adversarial atmospheric attacks, such as fog, rain, low light, and dust conditions, is a challenging task with regards to computer vision. Moreover, the applicability of convolutional neural network-based object detection architectures in various weather-affected night-time thermal scenes has not been extensively reported in recent and past literatures. The extraction of region of interest through anchors from each multi-resolution feature map (FM), either shallow or deep, suffers from several issues in adverse weather-degraded scenarios. Our proposed architecture, namely adverse weather-affected night scene restorator cum detector net (AWRDNet), focuses on the process of recovering such adverse weather-degraded video frames to restored frames through deeper convolutional layers. Further, our network reduces the time-consuming generation of pre-defined anchors in each FM at a deeper de-convolution layer, which combines different scales and aspect ratios for anchor boxes from multiple sets to naturally handle objects of various sizes. Considering the multi-scale anchor boxes at multiple set, an anchor refinement strategy has been applied to reduce memory consumption. The performance of the AWRDNet architecture is evaluated using standard detection performance metrics over the Tripura University Video Dataset at Night Time (TU-VDN) dataset which contains objects with annotated bounding box of the image frame sequences, and the available PASCAL VOC 2007 2012 datasets, and ZUT thermal dataset.

Keywords Adverse weather · Convolutional block · Deeper layers · Adaptive max pooling · TU-VDN · Object detection

1 Introduction

Under adverse weather conditions through atmospheric particles, the thermal infrared radiation signal must travel from the target to the camera detector sensor. Therefore, most signals can be altered or can lose their key characteristics along the way because of absorption and scattering by medium aerosols [1, 2], which produces blurry effects in the scenes. In case of visual sensors, the effects of different adverse atmospheric particles yield different degraded scenarios. Several methodologies have been developed for the restoration of different degraded scenarios, such as scattering model and dark channel prior. In thermal sensors, the different types of degradation caused by different atmospheric particles are indistinguishable, i.e., the effects are similar to the blur effect. Consequently, using thermal sensor-based frames for designing a single model to handle degraded blur thermal scenes for restoration via deblurring is advantageous. This encourages us to design a novel deeper convolution network, which is briefly elaborated in Section IV. Therefore, we address the problem of generating object detection system over atmospheric degraded scenarios. So far, many methods have been studied in computer vision for the above purpose, such as deep learning approaches [3, 4]. However, these deep approaches that are specially designed for producing high-resolution image are not for object detection purposes, and they rely on the context of small image patches. Therefore, we

✉ Mrinal Kanti Bhowmik
mrinalkantibhowmik@tripurauniv.ac.in;
mrinalkantibhowmik@tripurauniv.in

Anu Singha
anusingh5012@gmail.com

[1] Department of Computer Science and Engineering, Tripura University (A Central University), Suryamaninagar, Tripura 799022, India

[2] Department of Computer Science and Engineering, SRM Institute of Science and Technology, Ghaziabad, Delhi-NCR 201204, India

use the contextual information spread over the whole image to restore the atmospheric degraded images for more accurate object detection even in adverse atmospheric realistic scenes.

However, there is still a scarcity of video dataset object detection tasks that provide balanced coverage in weather-degraded outdoor scenes, especially at night. A satisfactory solution for night-time is necessary because darkness causes major safety problems because of collision of objects [5]. Furthermore, for detecting objects, far infrared cameras enable robust object detection irrespective of the atmospheric conditions because the effect of bad atmosphere decreases with the increase in spectrum wavelength [2]. To the best of our knowledge, object detection under adverse atmospheric conditions with night vision is very rare. Therefore, we provided a newly generated night-time dataset for detecting objects, which is briefly discussed in Section III. However, there have many key issues related to object detection at night [6, 7], such as flat and cluttered backgrounds.

There are two approaches to real-time detection of objects; two-stage [9, 10] and single-stage [13, 14] detectors. Multi-resolution FMs in single-stage were successfully used in object detector networks in [15, 18] for the problem of object detection; however, they have the following limitations. (i) Affect of shallow layers over smaller scale objects: The initial shallow layers of the single-stage object detector networks are not useful in weather-degraded scenes because of its lack of efficiency in the restoration task. As consequences, the performance in detection of long distance based smaller scale objects from sensors is becoming poor. Because the quality of outdoor images is affected by intensity, colour, polarization, and coherence of the light source due to scattering by medium aerosols [3, 4]. As a result, the contrast of the images is directly affect the shallow layers. (ii) Low resolution of big scale objects: The well restoration at deeper layers works efficiently, which are specially for detecting bigger objects; however, it suffers from low-resolution affects as there is no direct relationship exists among adjacent pixels on the output feature map. To mitigate the problem, we utilized the up-sampled feature map.

The above-mentioned approaches focus on detecting objects in everyday realistic scenes containing common objects. To the best of our knowledge, till date, no network has been proposed for outdoor scenes that are affected by several atmospheric conditions. In such complex scenes, the region proposal methods that typically rely on inexpensive features, such as selective search (SS) [12], which greedily merges super pixels based on engineered low-level features, is not suitable because of the blurred, flat, or cluttered textual nature of the scenes. As shown in Fig. 1a, it is difficult to regress region proposals to precisely surround the object (for instance, the pedestrian shown in Fig. 1 sample frame which is collected from the TU-VDN [6, 7] dataset under rainy condition at night-time.). There are many unwanted small region proposals generated by SS, which is because super pixels over flat or cluttered regions do not merge. In contrast, as shown in Fig. 1b, we adopt the strategy of pre-defined anchor boxes with several aspect ratios and scales, which is similar approach as in faster-RCNN [11]. These pre-defined anchors detect the pedestrian with better initialization that is the strength of single-stage networks which reflect the drawbacks of the two-stage region proposal networks.

The primary contributions of this study are summarized as follows. (i) In this study, we describe a comprehensive thermal video dataset of outdoor night adverse weather scenes, namely TU-VDN which consists of 60 video sequences and bounding box-based ground-truth of 22,030 number of frames. (ii) We introduce a single-stage AWRDNet architecture for detecting objects more accurately over degraded atmospheric scenes. (iii) We generalize AWRDNet architecture (i.e., PART-A) for restoration of degraded frames before object detect task. (iv) To achieve high detection accuracy, at anchor generation phase, we create anchor boxes with several aspect ratios and anchor scales only on deeper de-convoluted restoration FM. We also adopted an anchor refinement strategy to consume lower memory. (v) Experimental analysis on TU-VDN dataset reveals that the performance accuracy in low-light or rainy conditions is higher than that in dusty or foggy conditions. The ablation experiments also disclose that AWRDNet in a single-stage network outperforms than the two-stage networks. Furthermore, experimental analysis extended on PASCAL VOC dataset, and a comparative assessment on the ZUT thermal dataset also done.

The remainder of this study is organized as follows. In Sect. 2, we present a brief survey over deep architectures-based object detection. In Sect. 3, we describe the brief dataset design, ground truth generation, and statistics. We define the problem in Sect. 4 and describe the proposed architecture in Sect. 5. In Sect. 6, we present a complete evaluation of the captured dataset, followed by a discussion of the experimental results of the proposed architecture and a performance comparison with state-of-the-art approaches. In Sect. 7, we perform complexity analysis. Finally, in Sect. 8, we present the conclusions of this study and discuss future work.

2 Related work

Recent advances in object detection research areas in computer vision are driven with rapid successes of deep learning and convolutional neural network (CNN). Zhao

(a) SS algorithm outputs (b) Pre-defined anchors with different aspect ratios and scales

Fig. 1 Comparison of single-stage anchors and SS outputs. **b** Grid over 16 subsampling ratio

et al. [8] presented an overview of modern object detection approaches. There are two approaches to real-time detection of objects; two-stage and single-stage detectors. The two-stage detectors represented by the region CNN (RCNN) family [9–11] usually attain an accurate yet relatively slow performance. First, these detector families regress the pre-defined anchors with the help of hypothesis region proposals, such as region proposal networks (RPNs) [11] or selective search (SS) [12], and then run a classifier on these proposed boxes, which are complex pipelines that slow down the process for optimization. In contrast, detecting objects using one-stage detectors [13–15] directly regresses the coordinates from the pre-defined anchors, which results in a significant improvement in detection speed.

Two-stage-based methods The Szegedy et al. [23] and region CNN (RCNN) [9] models use same network architecture, wherein Szegedy et al. [23] trained their model from random initialization, and RCNN [9] uses supervised ImageNet pre-training to get 30% more mAP. Hoffman et al. [19] trained RCNN through transfer learning for classes that have image labels. In [37], Li et al. proposed an efficient regression model based on a generic CNN-based classifier, called adaptive deep CNN (ADCNN). ADCNN has been separated into two parts according to the function of the convolutional layers. The first part is a strong feature extractor based on the generic CNN-based classifier (the pre-trained CNN), and the second part is a special CNN architecture used for location prediction. The proposed method to construct surveillance scene-specific over only two challenging tasks, i.e., pedestrian and vehicle detection. These methods are mainly used for classification and refining bounding box by regression; however, the object coordinates are not predicted. This problem can be addressed in the following manner. (i) The overfeat [24] method predicts the object coordinates for a single object. (ii) The multibox methods [25, 26]—generalize the overfeat method to predict multiple class-agnostic boxes. (iii) The disadvantage of these multibox approaches are that they do not share features between region proposal and detection networks. Spatial pyramid pooling networks (SSPnets) [20] and Fast-RCNN [10] were proposed to speed up RCNNs by end-to-end detector training on shared convolutional features. The popular region proposal approaches that are used for RCNNs are SS [12] and Edgeboxes [27] for object detection. At this point, Cheng et al. [40] generated initial object proposals by hierarchical super-pixels using a tree-organized structure. Then CNN has been learned to select only a few proposals via object refinement. To address the proposed problem, authors start with adopting super-pixel hierarchy (SH), which is a spanning tree image structure for efficiently generating hierarchical super-pixels. The bounding boxes enclosing the generated super-pixels are treated as initial object proposals. For more surveys over proposal algorithms follow an article by Hosang et al. [28]. The generation of region proposal using external algorithms [12, 27] as independent module besides detector network is time consuming. Therefore, a new faster-RCNN [11] was proposed to improve the quality of proposals by a RPN. It is constructed using some pre-defined anchors at each location on a regular grid. Chen et al. [38] introduced a network, namely, 'PDC-Net' on a two-stage base network Faster-RCNN. This network specially analyze the procedure of statistical dependency between object proposals and refined bounding boxes to calibrate incorrect object category prediction detection results. Jie et al. [39] also presented a framework utilizing fully convolutional networks (FCN) to produce high-level semantic object proposal to localize object positions. It trains an object/non-object binary classifier using an FCN on patches from images with annotated objects. The FCN can take an input image of arbitrary size and output a dense "objectness map" showing the probability of containing an object for each corresponding box region in the original image.

Single-stage-based methods It is heuristic that the higher accuracy of two-stage methods comes with two advantages: two step regression and relatively accurate features for detection. Moreover, this two-step regression makes the approaches slow. To reframe object detection with inference speed, the single-stage approaches are used. One of the very first method, You Only Look Once (YOLO) [13] is a single regression approach that predicts bounding box

co-ordinates and class probabilities straight from image pixels or FMs. It is fast (45 FPS); however, it still lags behinds in accuracy as compared to state-of-the-art detection systems. For better performance by keeping fastest property, another version of YOLO—YOLOv2 [14] was proposed by J. Redmon et al. with deeper network. For variants scale objects, another set of approaches [15–18] used different layers within a ConvNet to predict fast and more accurately. Using single shot multibox detector (SSD) [15] is one of those approaches. It used default boxes of different aspect ratios to scales on multiple scale layers. It is deeper and faster—59 FPS. A deeply supervised object detector (DSOD) [16] was also built upon the SSD framework, which produces a simple and efficient model for object detector from scratch. The multi scale-CNN (MS-CNN) [17] and deconvolutional single shot detector (DSSD) [18] apply the concepts of deconvolution on multiple scale layers in the ConvNet to increase the resolution of FMs before using the layers to learn region proposals and object detection.

3 Brief description of the TU-VDN dataset with newly generated bounding box ground truth

The existing well-known object detection datasets, such as PASCAL VOC [29], MS COCO [30], and ImageNet VID [31], consist of challenges, such as realistic scenes, which gather images of complex everyday scenes, movement type, level of video clutter, and so on. Thus, it is difficult to evaluate the robustness of an object detection method under atmospheric conditions because aerosols reduce the visibility of targets in a scene. Therefore, we designed a standard night-vision video dataset, which is based on several atmospheric-weather-degraded conditions and covers many real-world scenarios. The dataset video recording conditions, dataset information, key features, ground truth annotation (binary mask based) details, and related key features of the designed dataset are discussed in our articles [6, 7]. These articles are based on moving object segmentation, where ground truth annotations are purposely generated for foreground object detection, i.e., binary mask generation. In this study, we have generated a bounding box-based ground truth generation for object detection. The TU-VDN dataset provides a realistic diverse set of outdoor videos in night vision that consists of 60 video sequences under various atmospheric conditions. Each video clip was two minutes in duration, the number of frames per videos was 2500, and the total number of frames was 138,230.

3.1 Bounding box ground truth generation of salient objects on the created dataset

Ground truth generation of salient objects allow understanding the efficiency of object detection algorithms. The manual fixations of salient objects in the form of bounding box indicate its identity and provide detailed spatial and temporal information. To implement the protocol, each laboratory member is asked to free-view all the extracted frames of the video clips distributed to them and to fix the two co-ordinate points of the salient objects in one annotating frame per five frames using the "LabelImg" graphical image annotation tool. Most left corners ($X_{\min}$, $Y_{\min}$) and lower-most right corners ($X_{\max}$, $Y_{\max}$). Along with bounding box information outlining the salient objects, temporal information related to object class as presented in the corresponding frames for each video clips of the created dataset are also maintained in ".xml" file. We labelled the objects present in the frames in 13 classes. The overall statistics of the ground truth annotated frames are presented in Table 1.

3.2 Data augmentation

In deep learning classification, a common problem is adjusting the overfitting issue. There are several methods to reduce overfitting in CNN models. The best option is to get more training data because our TU-VDN dataset have only 22,030 ground truth frames, which is a comparatively small dataset. The generation of more data through data augmentation is more beneficial only for smaller datasets and is capable of reducing overfitting issues. In this study, we chose three data augmenting procedures; horizontal flipping, HSV color space, and sequence. *Horizontal flipping (HF):* An augmentation that horizontally flips the frame for classification tasks is easiest. However, performing the same augmentation for an object detection tasks also requires updating the bounding box. Figure 2 shows the changes of bounding boxes during random horizontal flipping which flips a frame along with ground truth salient objects with a probability of "$p = 0.5$". *HSV Color Space:* we can usually get better information from a HSV color space [39]. For instance, a frame sequence where thermal temperature adjustment during the maiden appearance of a moving object causes illumination type effects in the current video frame [7]. Figure 3b shows a frame where the temperature polarity changed by the maiden appearance of a moving object (Truck). In the RGB color space, the illumination-effected frame will have varied characteristics than previous frame in a video sequence, as shown in Fig. 2a, without illumination effected. In the HSV color space, the "hue" component of both frames is more likely

Table 1 Statistics of the bounding box ground truth frames of TU-VDN dataset

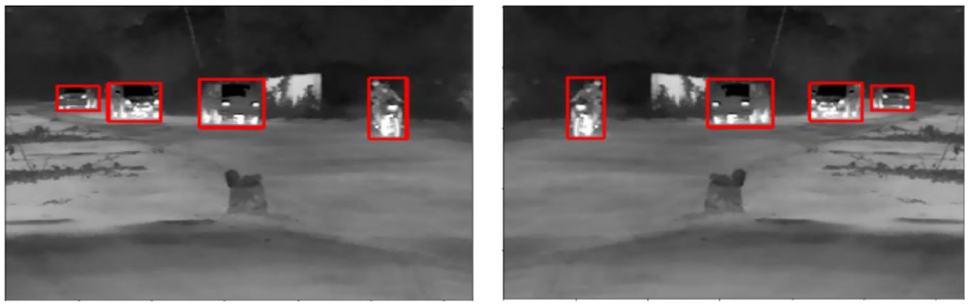
Atmospheric conditions	Number of annotated videos	Number of frames	Number of ground truth frames	Name of the classes											
				Pedestrian	Motorbiker	Bicyclist	Motorbike	Bicycle	Car	Bus	Truck	Mini Truck	Auto	Train	Rickshaw	Dog
Low Light	23	51,874	8771	17,267	1948	1934	1289	1635	1832	0	40	525	2242	471	65	444
Dust	16	37,439	7101	5464	4441	787	1095	378	5461	161	3784	645	1045	0	141	154
Rain	8	19,603	2073	2104	235	3	1	0	532	0	0	0	6	0	0	0
Fog	13	29,314	4085	2457	407	412	0	0	715	1	424	112	376	0	0	145
Total	60	138,230	22,030	27,292	7031	3136	2385	2013	8540	162	4248	1282	3669	471	206	743
Augmented data	60	138,230	144,121	27,292	28,124	25,088	26,235	26,169	25,620	4050	25,488	25,640	25,683	11,775	5150	18,575
Total number of annotated objects				274,889												

to be similar, thereby indicating that the primary color should not change much. Therefore, using only the "hue" component makes the data less sensitive to temperature polarity changes (illumination type effects). To characterize the HSV color space over temperature polarity changes in thermal frames, we used entropy to measure the contents, as shown in Fig. 4.

Sequence It is also defining a data augmentation that does nothing of its own characteristics; however, a combination of data augmentations can be applied in a sequence. The main purpose of this sequence is to increase the number of data and corresponding objects along with the advantages of HF and HSV. We attempted to balance the number of objects in each class using these three procedures through several permutations with parameters. The total number of annotated frames and objects after data augmentations are presented in Table 1.

4 Problem definition

More of the thermal infrared radiation signal can be lost along the way during traveling via adverse weather conditions which produces a degraded image [7]. Detection of accurate object under such degraded conditions is promising by existing state-of-the-art CNN-based object detection approaches [10, 11, 13, 15]. Therefore, we investigated on a deep learning structure on this problem before the beginning of detection strategies. The deep learning approaches with deeper convolutional layers can be restored of a degraded image [3, 4]. The receptive field of each deeper layer plays a vital role to analyze local features in such degraded images. The idea of receptive field in shallow layers is to extract local features and then combine them to make more complex and concentrate patterns in deeper layers. Consider that we are extracting just one feature per convolution layer, as shown in Fig. 5. The convolution layers are FM_0 (intensity-based input image region or feature map (FM)), FM_1 (first output FM), FM_2 (second output FM) with stride of 1 and convolutional kernel size 3×3. The receptive field of an image region in a convolution layer would be the cross section of the previous layers with kernels. Thus, the receptive field at coordinate (0, 0) of first FM ($FM_1(0,0)$) is the cross section of local region square $FM_0(0{:}2, 0{:}2)$ over 3×3 convolution kernel 1 (K_1). The receptive field of $FM_2(0,0)$ will be crossed of $FM_1(0{:}2, 0{:}2)$ which itself receives inputs from $FM_0(0{:}4, 0{:}4)$. Therefore, we have stacked 3×3 kernel in two intermediate convolutional layers, which produces results similar to that obtained using a single kernel of size 5×5. Consequently, three convolutional layers would give us an effective size of 7×7 kernel and so on.



(a) Original frame (b) Horizontally flipped frame

Fig. 2 Sample frames from TU-VDN dataset along with bounding box ground truths

Fig. 3 Sample thermal frames from TU-VDN dataset where temperature adjustment causes illumination type effects. **a** A normal frame sample, **b** temperature polarity changed next frame, **c** HSV space on the normal frame sample, and **d** HSV space on the temperature polarity changed next frame

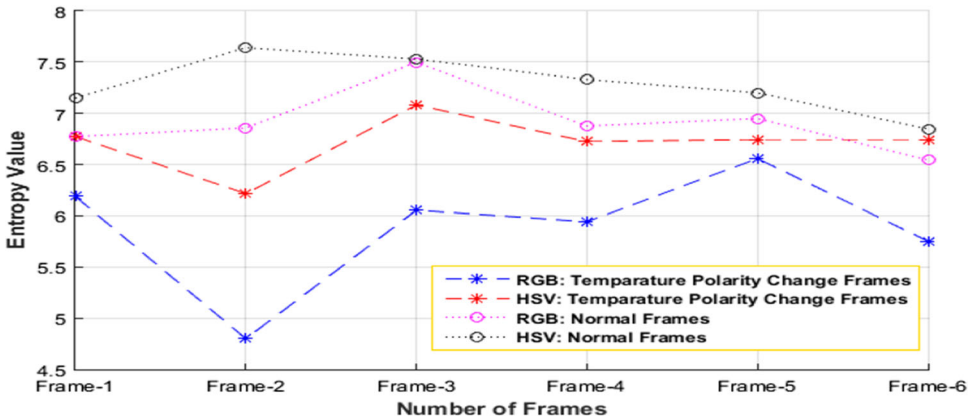


Fig. 4 Characterization of thermal illumination affected frames from the TU-VDN dataset through entropy value. The higher entropy values of illumination effected thermal frames on the HSV color space over RGB color space indicates an image with adequate details of information in terms of better quality. In case of normal frames, the differences between RGB and HSV color space entropy values are lesser but higher than illumination effected frames

Proposition *The deeper convolutional layers produce better quality of restorated image.*

Proof As shown in Fig. 5, the add up of convolutional FMs with original images will result in restorated images RI_1 and RI_2, where the value (v) of pixels in deeper FMs are lesser than shallow layers as follows:

$$v\left(\widehat{\mathrm{FM}_0(x,y)}\right) > v(\mathrm{FM}_1(x,y)) > v(\mathrm{FM}_2(x,y)) \tag{1}$$

where $\widehat{\mathrm{FM}_0}$ is a normalized image region, i.e., pixel values between 0 and 1.

Equation (1) holds the conditions because convolution on a receptive field over another receptive field will always produce lesser resultant values into the output cells (in case of a normalized image).

Considering quality measures from peak signal-to-noise ratio (PSNR), Eq. (1) can be expressed as follows:

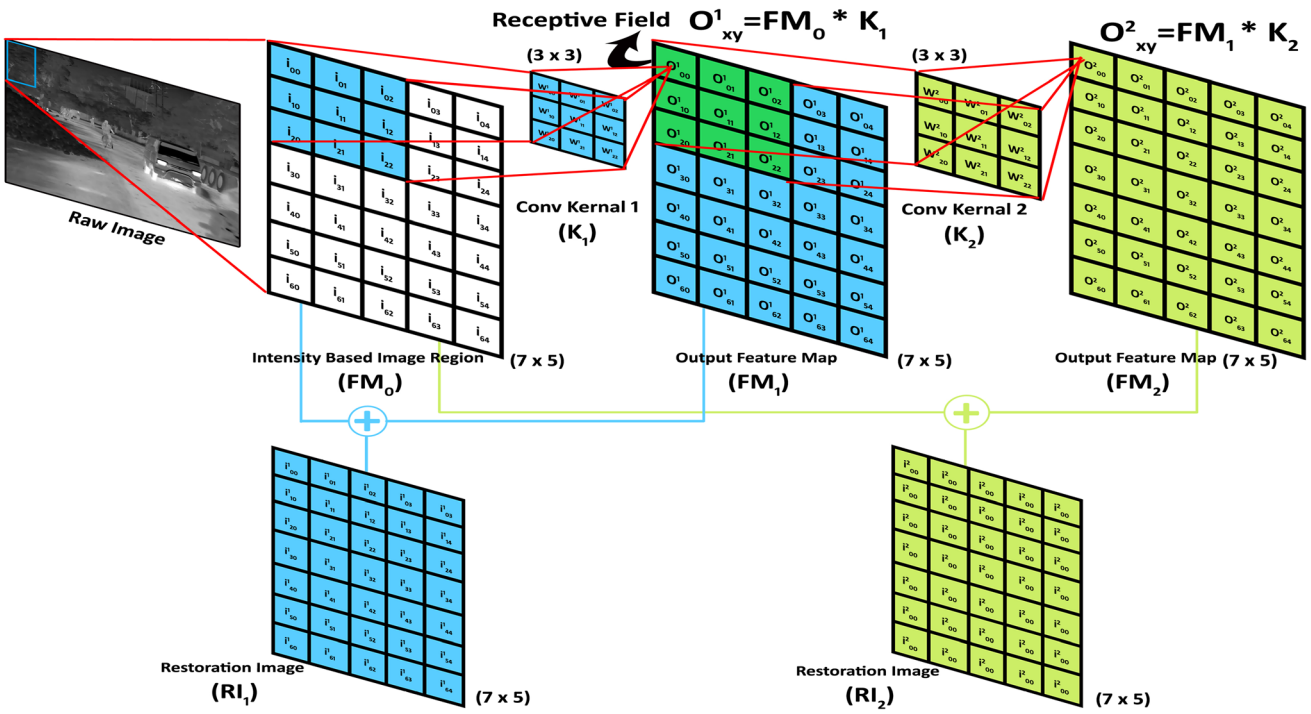


Fig. 5 Deeper convolutional layers for better restorated feature map

$$v(\hat{FM}_0(x,y)) > v(FM_1(x,y)) > v(FM_2(x,y))$$

$$\text{PSNR}_1 \quad \text{PSNR}_2$$

$$\therefore\ 20\log_{10}(\hat{FM}_{0\,\max}) - 10\log_{10}[\frac{1}{mn}\sum_{x=0}^{m-1}\sum_{y=0}^{n-1}\{\hat{FM}_0(x,y) - FM_1(x,y)\}^2]$$

$$< 20\log_{10}(\hat{FM}_{0\,\max}) - 10\log_{10}[\frac{1}{mn}\sum_{x=0}^{m-1}\sum_{y=0}^{n-1}\{\hat{FM}_0(x,y) - FM_2(x,y)\}^2]$$

$$\Rightarrow 0 - 10\log_{10}(mse_1) < 0 - 10\log_{10}(mse_2) \quad \because \hat{\text{FM}}_{0\,\max} = 1$$

$$\Rightarrow \log_{10}(mse_1) < \log_{10}(mse_2) \tag{2}$$

Equation (2) holds the mentioned condition ("<") because the mean square error (mse) is estimated based on difference between original and restoration FMs. The pixel values of restoration FM over the deeper convolution layers will result in lesser than shallow convolution layers, as expressed in Eq. (1), i.e., $v(\text{FM}_1(x,y)) > v(\text{FM}_2(x,y))$. Therefore, the mse value of first convolutional layer-based restoration FM will be lesser than second convolutional layer-based restoration FM, and correspondingly logarithm value of first convolutional layer-based restoration FM will be lesser than second restoration FM. For example, the value of mse_1 between FM_0 and FM_1 is 0.4 and the value of mse_2 between FM_0 and FM_2 is 0.5 (the pixel values will be in between 0 and 1, and convolutional operation will also resultant lesser than 1 because the original image is normalized) in Eq. (2) as follows:

$$\begin{aligned} &\log_{10}(0.4) < \log_{10}(0.5) \\ &\Rightarrow -0.397 < -0.301 \\ &\Rightarrow \text{PSNR}_1 < \text{PSNR}_2 \end{aligned} \tag{3}$$

From Eq. (3), we can say that the deeper convolutional layer will produces better quality of an image.

5 Proposed architecture

In this section, the proposed AWRDNet will be presented, as shown in Fig. 6. The AWRDNet approach consists of three portions. (Sect. 5.1) The first portion is based on a feed-forward convolutional network that produces better restorated images through deeper convolutional layers. We generalize AWRDNet architecture PART-A for restoration of degraded images before object detect task (Sect. 5.1.1). (Sect. 5.2) The second portion generates a fixed-size collection of pre-defined anchors on the de-convolutional FM of deeper convolutional layers, and refinement of anchors. (Sect. 5.3) Finally, for each object anchor, extracts a fixed-length feature vector which is fed into a sequence of fully connected layers that subdivided into two sibling output layers: one that estimates SoftMax operation to produce probabilities over "$P + 1$" object classes where "plus 1" is for a background class, and another branch that produces

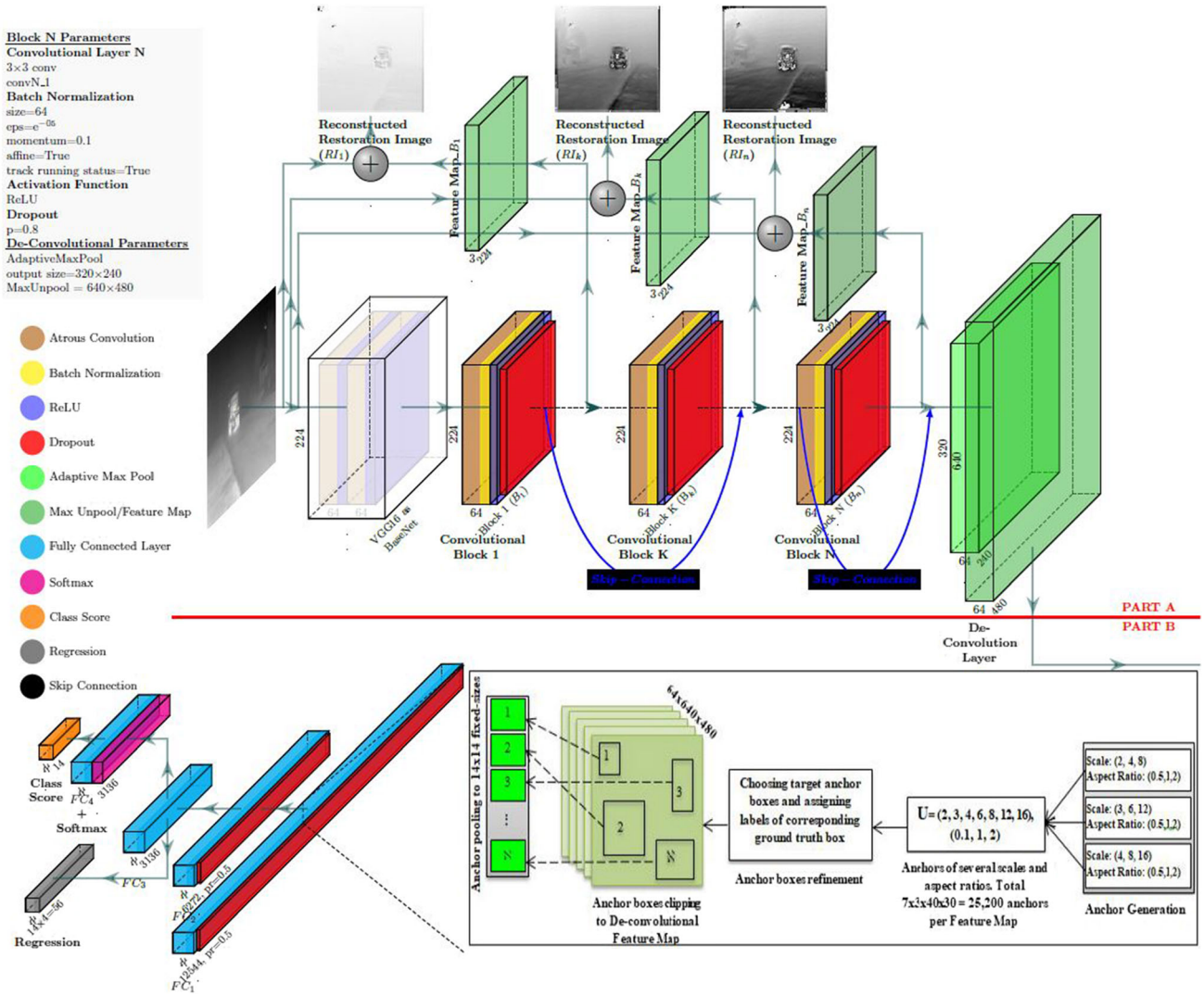


Fig. 6 Schematic layout of the proposed AWRDNet architecture. Restorated FM is produced through *N* number of convolutional blocks, and final FM prepared by deconvolution procedure for setting up pre-defined set of anchor boxes for object detection and labeling tasks

4 (four) real valued bounding box position numbers for each of the "P" object classes.

5.1 Deeper convolutional layers for restoration of adverse atmospheric degraded thermal frames

The atmosphere influences the visibility through aerosols; the type of infrared camera that is used and the waveband in which the camera operates are also important because of the following reasons. (i) The particles size exceeds the wavelength in the visible portion of the electromagnetic spectrum (0.4 to 0.74 μm), attenuation by atmospheric aerosols is independent of the wavelength. (ii) As the wavelength increases, attenuation becomes less of an issue. Wavelengths in the far-infrared region (5–14 μm) exceeds those of other infrared wave bands (0.74–5 μm); thus, impact of particles on far-infrared waves is relatively insignificant. (iii) Far-infrared wavelength is higher than other infrared wavebands; however, the particles size of fog and rain much higher than far-infrared waveband length, especially rain water droplet particles size 500–5000 μm where fog water droplet is only 0.5–80 μm [33]. Therefore, the degradation of night images even in thermal frames still there. In last one decay, numerous numbers of deep learning approaches has been developed for realistic scenes-based object detection; however, the focus on adverse weather affected realistic real-world scenes is still lacking. For restoration-based image FMs generation, Fig. 6 shows a deep convolutional network where initial convolutional layers under a BaseNet, Convolutional Block-1, Block-K, Block-N, and De-Convolution.

First, the proposed network processes the whole frames through 2 convolutional layers and 64 filters from VGG16 network as base network with pre-trained ImageNet and produce a FM of 64 × 224 × 224 tensor. The rest of convolutional layers contains under "N" number of deep convolutional blocks, where each block consists of a convolution layer, batch normalization, activation function followed by dropout. Our emphasis is on restoration of degraded image through each block; thus, we have skipped the pooling operation per block to avoid dimension reduction in output FMs. We avoid the pooling operation because it reduces the memory consumption and decreases the resolution. Our main motive is generation of restorated FM before beginning of object detection tasks; thus, the convolution-AdaptivePooling operation also requires deconvolution operation to increases the resolution of FMs [18].

Atrous convolutional layer In this study, the atrous convolutional layer consists over "N" number of blocks.

The reason behind replacement of normal convolution with atrous convolution is exponential expansion of reception field which is support exponential expansion of the receptive field without loss of resolution. It is applied to input feature map ($F:Z^2 \rightarrow R$ be a feature map discrete function) with definite gaps in the kernels [$k:\Omega_r \rightarrow R$ be a kernel of size $(2r+1)^2$]. Atrous convolution can be formulated as [56]

$$(F *_a k) = \sum_i \sum_{aj} F(i)k(aj) \tag{4}$$

where 'a' be a atrous factor. If atrous rate is 1, it means the convolution kernel is normal, and if the atrous rate is 2, then there is a skip of one pixel per input. Increasing the stride reduces the dimension of the output. A 2 × 2 atrous convolution has the same receptive field as a 3 × 3 un-atrous convolution.

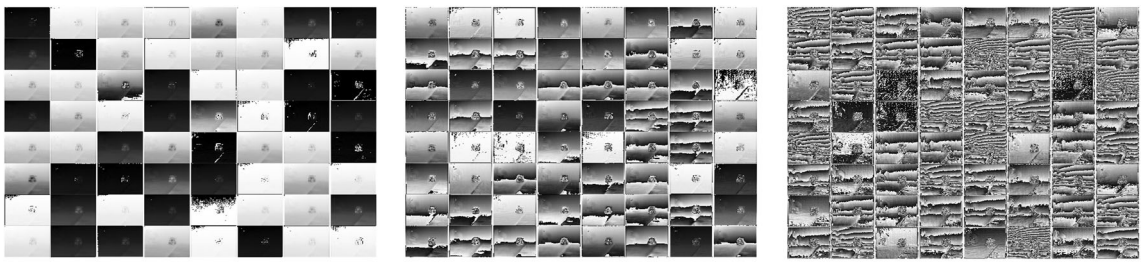
Finally atrous convolution is a mathematical operation "*," namely, convolution that takes two inputs such as image feature tensor of dimension "batch_size × d × height × width" and a kernel of dimension "$d \times k_{\text{height}} \times k_{\text{width}}$". We use "$d$" kernels ($d$ = 64 in our case) of the size "$d \times 2 \times 2$", where a kernel operates on "2 × 2" receptive field across "d" channels. The network takes an interpolated degraded image of size "3 × 224 × 224" as input feature, and outputs will be volume of dimension $O = (O - F + 2P)/S + 1$, where "O" (O = 224 in our case) is the input size, "F" (F = 2 in our case) is the receptive field size, "P" (P = 1 in our case) is the padding to fit receptive field perfectly to the input image/FM, and "S" (S = 1 in our case) is the stride, i.e., number of pixels shifts over the input image/feature matrix.

The convolution of a degraded image with "d" number of kernels can perform several random operations, which are advantageous in decoding several undetectable salient features. Figure 7 shows various convoluted images after applying different types of d = 64 number kernels. In the initial block = 1, we can see more clear scenes of a night thermal foggy input image (as shows in Fig. 7) and few output images of several kernels become a usual input or darker than input. Further, the next intermediate block = K, we can visualize more abstract clearer forms. The deeper block = N that provide more and more concrete forms of a degraded image.

Batch normalization To increase the stability and speedup learning network, we normalize the convolution layer through two trainable parameters shifting (γ) and scaling (β), which are learned during training along with the original parameters of the network. Therefore, batch normalization allows each layer of a network to learn by itself a little bit more independently of other layers. It normalizes the output of a previous convolution layer by subtracting the mean ($\mu_{\mathscr{B}}$) of mini-batch $\mathscr{B} = \{o_1, o_2, \ldots, o_d; o_i$ is a channel output$\}$ and dividing by the batch standard deviation ($\sigma_{\mathscr{B}}$). Therefore, batch normalization can be expressed as follows:

$$\hat{o}_i = \frac{o_i - \mu_{\mathscr{B}}}{\sqrt{\sigma_{\mathscr{B}}^2 + \varepsilon}} \tag{5}$$

where



(a) Block = 1 (b) Block K= 5 (c) Block N = 10

Fig. 7 Various convoluted images using 64 numbers of kernels over several deeper blocks

$$\mu_{\mathcal{B}} = \frac{1}{d}\sum_{i=1}^{d} o_i \sigma_{\mathcal{B}}^2 = \frac{1}{d}\sum_{i=1}^{d}(o_i - \mu_{\mathcal{B}})^2$$

$\varepsilon = e^{-05}$ constant to avoid complex value.

Batch normalization also reduces overfitting because it has slight regularization effects. Therefore, scale and shift the normalized batch to obtain the output of the layer as follows:

$$o_i = \gamma \widehat{o_i} + \beta \tag{6}$$

where γ and β are the scaling and shifting factors, respectively.

Non-linearity To introduce non-linearity in our system without making a significance difference to the output of normalized convolution, we used ReLU activation function $f(o_i) = \max(0, o_i)$. The ReLU activation function does not changes much of the normalized convolution outputs except all the negative activations to 0, since the real-world data would want our network to learn would be non-negative linear values.

Dropout Data normalization has slight regularization effects that can reduce overfitting. Therefore, if we use batch normalization, we should use less dropout ($pr = 0.8$ in our case), which is a worthy because it will not lose a lot of information. However, we should not depend only on batch normalization for regularization; we should better use it together with dropout.

Skip-connection Now, the each convolutional block will be gradually reconstruct the feature maps for restoration of degraded maps through high level information. The feature maps after each block concatenate via skip connection with the corresponding feature maps from the previous block to avoid losing pattern or spatial information. The concatenation can fuse the low and high level information of the feature maps, and enhance the perception ability to degraded feature maps.

In skip connection operation, the skip connection explicitly concatenate the feature maps generated in previous block (B_{k-i}) with current block (B_k) feature maps. Let $\in_{B_k}^{B_{k-i}}$ be concatenate layer. The convolutional (normal or atrous) feature map $\complement^{B_{k-1}}$ in the preceding layer is upsampled by a scale factor φ_s where $\varphi_s = 1$ which will keep the same dimension (as our model having same dimensions over the blocks) of the $(k-1)$th block by a factor of 1 and concatenate it with an previous block convolutional feature layer $C^{B_{k-i}}$ where i be the number of skipped layers from the concatenate layer. It can be formulated as

$$\in_{B_k}^{B_{k-i}} = \complement^{B_{k-1}} * \varphi_s \oplus C^{B_{k-i}} \tag{7}$$

After concatenation we reduce the channel numbers again to 64 from 128.

Consequently, each block will consist of convolution, batch normalization, non-linearity, and dropout operations. The deeper ConvNet block will use, the better restoration image can get back. In the study, we are using $N = 10$ number of blocks. For the restoration image, once image/FM details are predicted in each block, they are added back to the input degraded image to provide the restoration image, as shown in Fig. 6. We have analyzed the quality of these restoration images, as shown in Fig. 8. We have seen that after each convolution block, it produces better PSNR values, thereby indicating that the deeper convolutional layers produce better restoration images, which also proved in Proposition.

After Block N of size $64 \times 224 \times 224$, we have used adaptive max pooling of desire output size of $64 \times 320 \times 240$. The adaptive output FM again max unpooled to $64 \times 640 \times 480$ as original dataset frames size to clipping pre-defined anchor boxes or ground truth bounding boxes.

5.1.1 Formulation of a sub-architecture for restoration

Figure 9 shows a separate sub-architecture to evaluate the restoration work on Part-A from proposed AWRDNet model. The upscaled de-convoluted layer first reduce a number of channels as desire size ($I'_{480\times640\times3}$) to estimate error. Our goal is to recover an image I' which is as possible clear and concrete visible image. Since our capture original images are by-default degraded, there is no reference images to take as ground-truth for loss estimation. Therefore, we applies a trick here: restorated image I'_i on first iteration (a.k.a. previous) used as reference image and restorated image I'_j on second iteration (a.k.a. next) used as current image to calculate loss function. Then, all the filters weights and biases are to be optimized via Adam approach.

Loss function The main purpose of image reconstruction is not only to improve enhance visibility, also enhance edge-texture information, maintain color-structure of the image. In order to evaluate this method, PSNR is used and mean square error (MSE) for the loss function as

$$L = \frac{1}{N}\sum_{i=1}^{N}\left\| I'_i - I'_{j=i+1} \right\|^2 \tag{8}$$

where N is the number of training samples, I'_i represents the reconstructed image which is used as reference image, and I'_j denotes the reconstructed image on next iteration. The loss is minimized via Adam optimizer with the standard back propagation.

Using MSE as the loss function courtesies a high PSNR which is a widely used metric for evaluating reconstruction quality. The overall performance evaluation over different weather conditions is studied in Sect. 6.1. To provide a better visual understanding of the reconstruction images, typical results are shown in Fig. 10 under various atmospheric conditions.

5.2 Anchor boxes generation

We associate a set of different scales and aspect ratios to restoration based last de-convolution FM in Fig. 6. The FM is sub-sampled of 16 pixels which pooled our FM from 640 × 480 pixels to 60 × 40 pixels size. Now every pixel in the feature cells is 16 × 16 pixels along the x and y axes. At center of each feature cell, we will use anchor scales of (2, 4, 8) and aspect ratios of (0.5, 1, 2) to generate 9 anchor boxes width and height as follows:

$$\begin{cases} \text{anchor\_width} = \text{sub\_sample} \times \text{anchor\_scale} \times \sqrt{1/\text{aspect\_ratio}} \\ \text{anchor\_height} = \text{sub\_sample} \times \text{anchor\_scale} \times \sqrt{\text{aspect\_ratio}} \end{cases} \tag{9}$$

The aspect ratio values 0.5 and 2 in Eq. 9 indicate generalization of the horizontal and vertical anchor boxes with corresponding width and height. The horizontal anchor box will especially for vehicle type objects, and the vertical anchor box will for pedestrian type objects. The aspect ratio value of 1 will generate a square anchor box, which is suitable for objects with square shape positions. For example, human in sitting position or tiny animals like dog, cat. Therefore, anchor boxes will have shape of (9, 4) with four coordinate points at each cell of the FM:

$$\begin{cases} x_{\min} = x_{\text{center}} - \text{anchor\_width}/2 y_{\min} = y_{\text{center}} - \text{anchor\_height}/2 \\ x_{\max} = x_{\text{center}} + \text{anchor\_width}/2 y_{\max} = y_{\text{center}} + \text{anchor\_height}/2 \end{cases} \tag{10}$$

The attractiveness of existing anchor-based approaches is that they addressing multiple scales of "image or feature pyramids" and "pyramid of filters" [10, 20, 24, 34] or use of default boxes on multi scales of FM [15]. At shallow FMs, the default anchor boxes for small objects detection; at deeper FMs, the default anchor boxes for larger objects detection. These approaches are often useful but time consuming.

We are working only on a deeper restoration based de-convolutional FM; thus, the concept of multi scales anchor box generation over multi scale FMs is missing [15]. To capture the small or far objects and large or near objects, we are using another two set of different scale sets (3, 6, 12) and (4, 8, 16) with same aspect ratios, i.e., (0.5, 1, 2). The property of the proposed net based anchor boxes generation is translation invariant [11]. If we combine all three sets of multi scales, it will be 3 × 7 = 21 anchor boxes (3 aspect ratios × 7 multi scales after remove duplicates) per cell and total of 60 × 40 × 21 = 50,400 anchor boxes over 640 × 480 resolution FM with 16 sub-sampling. Generation of this anchor boxes approach is similar to the anchor boxes used in [11, 15]. However, we will find the index of all valid anchor boxes "ab" by applying conditions that should be followed as $\text{ab}(:,x_{\min}) \geq 0$, $\text{ab}(:,y_{\min}) \geq 0$, $\text{ab}(:,x_{\max}) \leq 640$, and $\text{ab}(:,y_{\max}) \leq 480$ where ":" indicates all anchor boxes together.

Anchor boxes refinement Based on the ground truth boxes, the bounding box regression from an anchor box to a nearby ground truth "gt" box parameterizations of the four coordinates as follows [9]:

$$\begin{cases} dx = (gt\_x_{\text{center}} - ab\_x_{\text{center}})/\text{anchor\_width} \\ dy = (gt\_y_{\text{center}} - ab\_y_{\text{center}})/\text{anchor\_height} \\ dw = \log(gt\_\text{width}/\text{anchor\_width}) \\ dh = \log(gt\_\text{height}/\text{anchor\_height}) \end{cases} \tag{11}$$

and the target boxes "tb" for loss estimation after each epoch will be

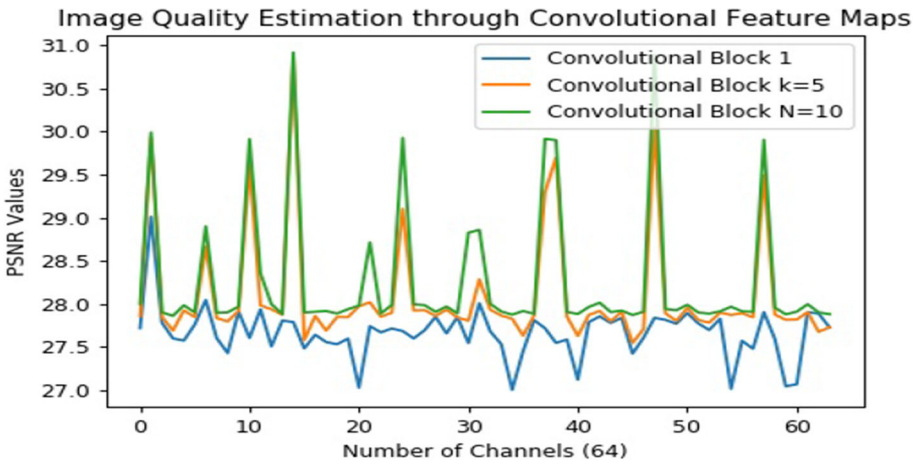


Fig. 8 Analysis of restorated images using peak-signal-to-noise (PSNR) over convolutional blocks in our AWRDNet network

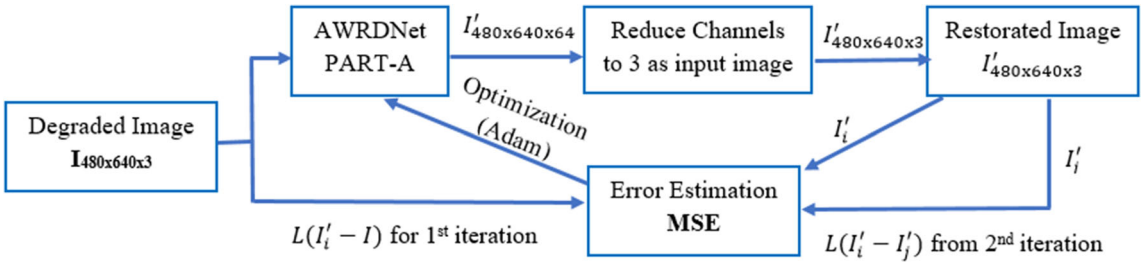


Fig. 9 Schematic flow diagram for restoration images. I'_i: first (previous) iteration reconstructed image where i = 1, 2, 3, … I'_j: second (next) iteration reconstructed image where j = 2, 3, 4, … L is the loss function

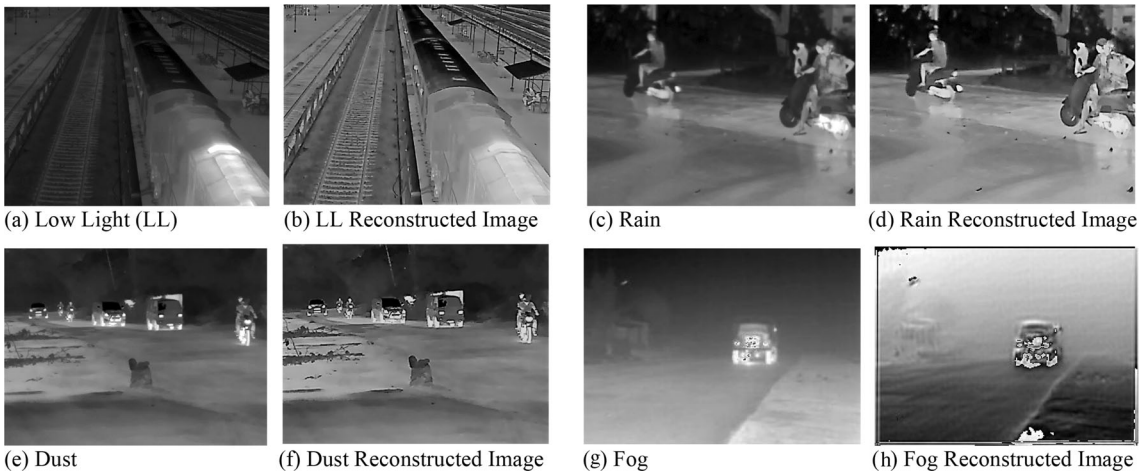


Fig. 10 Typical reconstruction results over various atmospheric conditions in our created night time dataset

$$\mathrm{tb} = [\mathrm{d}x, \mathrm{d}y, \mathrm{d}w, \mathrm{d}h] \tag{12}$$

For each valid anchor box, the estimation of intersection-over-union (IoU) with each ground-truth object will be

$$A(\mathrm{IoU}) = (X2 - X1) \times (Y2 - Y1) \quad \text{if} \quad X1 < X2 \,\&\, Y1 < Y2 \tag{13}$$

where $X1 = \max(gt(:, x_{\min}), ab(:, x_{\min})); X2 = \min(gt(:, x_{\max}), ab(:, x_{\max}))$; $Y1 = \max(gt(:, y_{\min}), ab(:, y_{\min})); Y2 = \min(gt(:, x_{\max}), ab(:, x_{\max}))$; 'A' indicates area under a box.

Therefore, the final areas for anchor boxes and bounding box ground truth objects are as follows:

$$\mathrm{IoU}(ab, gt) = \frac{A(\mathrm{IoU})}{(A(ab) - A(gt))} \tag{14}$$

where

$$A(ab) = \{ab(:, x_{\max}) - ab(:, x_{\min})\} \times \{ab(:, y_{\max}) - ab(:, y_{\min})\}$$
$$A(gt) = \{gt(:, x_{\max}) - gt(:, x_{\min})\} \times \{gt(:, y_{\max}) - gt(:, y_{\min})\}$$

We need to find the highest IoU (max_IoU) and index (max_idx) for each anchor boxes to its corresponding ground truth. Assignment of labels—index to all the relative anchor boxes "rab" (box coordinates range between 0 and 1) or target boxes to a list of region of interest those will have maximum IoU greater than positive threshold (pos_t = 0.4, neg_t = 0.1 in our case) as follows:

$$\begin{cases} \mathrm{RoI} = \mathrm{RoI} + rab \\ \mathrm{tRoI} = \mathrm{tRoI} + tb \\ \mathrm{pos\_idx} = \mathrm{pos\_idx} + \mathrm{length(RoI)} \\ \mathrm{labels} = \mathrm{labels} + gt\_class[\mathrm{max\_idx}] \end{cases} \quad \text{if} \quad (\mathrm{max\_IoU} \geq \mathrm{pos\_t}) \tag{15}$$

where RoI, tRoI, pos_idx, neg_idx, labels initially an empty list.

$$\begin{cases} \mathrm{RoI} = \mathrm{RoI} + rab \\ \mathrm{tRoI} = \mathrm{tRoI} + tb \\ \mathrm{neg\_idx} = \mathrm{neg\_idx} + \mathrm{length(RoI)} \\ \mathrm{labels} = \mathrm{labels} + 0 \end{cases} \quad \text{if} \quad (\mathrm{neg\_t} < \mathrm{max\_IoU} < \mathrm{pos\_t}) \tag{16}$$

Clipping anchor boxes to de-convolutional FM Clip the anchor boxes to de-convolution FM of size W = 640 and H = 480 as follows:

$$\begin{cases} x_{\min} = \mathrm{rab}(: x_{\min}) \times W \\ x_{\max} = \mathrm{rab}(: x_{\max}) \times W \\ y_{\min} = \mathrm{rab}(: y_{\min}) \times H \\ y_{\max} = \mathrm{rab}(: y_{\max}) \times H \end{cases} \tag{17}$$

The clipped anchor boxes to de-convolutional FM have different sizes, from where we can quickly get a list of corresponding anchor boxes with a fixed size. To this purpose, anchor pooling layer uses adaptive max pooling to transform the features inside any valid anchor box into a small FM with a fixed size of 14 × 14. This adaptive max pooling operation is applied for each anchor box to each channel ($d = 64$ in our case) of FM which is a good pyramid of resized boxes for each anchor box.

5.3 Classification and regression

As shown in Fig. 6, the resized anchor boxes then mapped to corresponding feature vectors, those are going through four fully connected layers (FCs). The first two FC layers goes through ReLU activation and dropout with pr = 0.5 and reducing feature vector dimension of 12,544 to 6272 and 6272 to 3136, respectively. The third FC branch out to a classification head and a regression head. The classification head operated over fourth FC followed by SoftMax to produces output class confidence scores. The confidence scores for "$P = 14$" object categories as $C = \{c_1, c_2, \ldots, c_p\}$. The regression head produces 14 × 4 = 56 offsets per class regressed bounding box (bbox). Finally, the loss value will be estimated of sums up the cost of classification head and bounding box regression head as follows:

$$\begin{aligned}\text{Loss} &= \text{CrossEntropyLoss}(\text{labels,C}) + \text{SmoothL1Loss}(\text{tRoI} + \text{bbox}) \\ &= -\frac{1}{\aleph}\sum_{i=1}^{\aleph} labels_i \times \log(c_i) \\ &+ \frac{\lambda}{\aleph}\sum_{i=1}^{\aleph}\sum_{j\in(x_{\min},y_{\min},x_{\max},y_{\max})} L_1^{\text{smooth}}\left(\text{smooth}_{ij} - \text{tRoI}_{ij}\right)\end{aligned} \tag{18}$$

where $\aleph$: total number of anchor boxes entered in FC layers as input feature vectors; λ: a balancing parameter, $\lambda = 1$ so that both classification and regression terms are roughly equally weighted.

5.4 Training and testing parameters

We pre-trained the proposed model over the VGG16 network as base network on the ImageNet multiclass competition dataset [31]. For pre-training, only two convolutional layers have used to reduce the large dimensional original frames to a VGG16 based small dimension (224 × 224) for learning speedup and less memory consumption. We trained the proposed network model for about a week for about 500 epochs on the training and validation datasets from TU-VDN. Throughout training, a batch size of eight was used, and the schedule learning rate slowly raises from $1e^{-4}$ to $1e^{-2}$ in the following order: $1e^{-4}$ for first 200 epochs, $1e^{-3}$ for next 200 epochs, and finally $1e^{-2}$ for 100 epochs. If we start at a high learning rate our network model often diverges due to unstable gradients.

We cannot use ground truth bounding boxes during testing; thus, the non-maximum suppression (NMS) over the proposed model produces bounding boxes with IoU threshold and confidence score threshold. The generated bounding boxes are highly overlapped with each other which can reduce the redundancy based on their threshold parameters.

6 Experimental evaluations and discussions

This section is divided into three subsections. First: we investigate the qualitative evaluation of restoration images, Second: we analyze the performance of the proposed network model, namely, AWRDNet over our night thermal dataset, namely, TU-VDN, which consists of four atmospheric conditions (low-light, dust, rain, and fog); those are realistic scenarios that are typically encountered in practice. We fine-tune the resulting proposed model using the "Adam" optimizer. Third: we compare and analyze the different proposal or anchor generation approaches to our proposed model on the TU-VDN dataset. Fourth: on this dataset, we compare the proposed network model against existing state-of-the-art twostage approaches such as fast-RCNN [10], faster-RCNN [11], G-RCNN [56], and single-stage approaches such as YOLO [13], YOLOv4 [55], YOLOR [54], SSD [15]. In Sect. 6.2, assessment of the proposed model using a widely popular realistic scene-based object detection dataset: PASCAL VOC [29] is done. The results are reported with respect to performance metrics: "mAP" to describe the detection accuracy, and a graphical interpretation based "recall-precision" graph. In last part (Sect. 6.3), evaluation and comparative assessment has conducted on ZUT [57] thermal dataset using F_1-measure.

6.1 Evaluation on the TU-VDN dataset over newly created bounding box ground truth samples

The models are trained on TU-VDN augmented dataset, as presented in Table 1, with total of 144,121 ground truth frames and 274,889 number of ground truth objects under 13 object categories. The whole dataset divided into two sets: one set consists of 80% of total data for training and another set consists of 20% of total data for testing. As well as, the training set sub-divided into 80–20% train-validation from training set data.

Qualitative evaluation of restorated images The TU-VDN degraded images over adverse weather conditions have been utilized to evaluate the performances of restoration methods. As shows in Fig. 11, the proposed AWRDNet_-PART-A yields the competitive average of PSNR values as compare to the state-of-the-art techniques. The state-of-the-art techniques used in this experiment along with our proposed model are sparse-coding based method (SC) [41], anchored neighbourhood regression (ANR) [42], super-resolution CNN (SRCNN) [43], graph convolutional network (GCN) [44], recurrent squeeze-and-excitation context aggregation net (RESCAN) [45], progressive recurrent network (PreNet) [46], spatial attentive network (SPANet) [47], and deep residual convolutional dehazing network (DRCDN) [48].

The average performance achieved by our proposed model are 39.15 dB (under low light condition), 39.08 dB (under rainy condition), 37.23 dB (under dusty condition), and 36.97 dB (under foggy condition), respectively. From weather point of view, the proposed model shows highest performance in low light condition followed by rainy condition. From the comparison with state-of-the-art methods, we have noticed that low light and rain conditions having good results over all methods than dust and fog conditions. The GCN (38.85 dB), RESCAN (39.10 dB), SPANet (38.19 dB), and PreNet (38.70 dB) giving higher performances than our model (38.08 dB). The reason behind these methods specially designed for de-raining challenges. Other than this, our model outperform than state-of-the-art methods at rest of conditions.

Analysis of the performance of AWRDNet model over different adverse conditions To demonstrate our contributions via analysis of our dataset using the proposed model, we present the performance evaluation in Fig. 12 in terms of detection evaluation recall-precision graph. To investigate the behavior of AWRDNet as a proposed model, we conducted several ablation studies, such as detection over single object data under adverse atmospheric conditions (as shown in Fig. 12a), detection over double objects data (as shown in Fig. 12b), and detection over multiple objects data (as shown in Fig. 12c). During testing on these ablation studies, we kept positive IoU threshold is 0.3 and negative IoU threshold in between 0.1 and 0.3, non-maximum suppression (NMS) IoU threshold is 0.6 along with confidence score threshold 0.6. On the single or double objects data, our model has maximum mAP $\approx$ 77% or $\approx$ 78% over low-light condition, and reduces in detection accuracy to $\approx$ 72% on multi-objects data scenes. Considering the atmospheric conditions, the low-light or rainy conditions are promising than dust or foggy conditions. The low-light giving the highest mAP in all types of ablation object data because as we know low-light is not an atmospheric condition, just included to analyze dark scenes under thermal camera. Besides that, under rainy condition, we realize the second highest detection accuracy values because as the aerosols size increases (the radius of the rain droplet is in the order of microns), less scattering is observed. Therefore, there is less loss of contrast, which reduces the false-negative analysis under convolutional layers. Whereas foggy and dust conditions loss more of contrast due to the dense of particles which affects in detection accuracy.

Analysis of anchor generation approaches over AWRDNet model To compare the single-stage based anchor generation approach and two-stage based proposal approach, we emulate these approaches by our AWRDNet model. Table 2 shows AWRDNet results when train and test using various region proposal or anchor approaches. For selective search (SS), we analyze about 300, 1 k, 2 k proposals, respectively, over two proposals refinement strategies, i.e., (I) proposal positive index if IoU $\geq$ 0.5 and negative index if 01 $\leq$ IoU $<$ 0.5, (II) proposals positive index if IoU $\geq$ 0.3 and negative index if 0.1 $\leq$ IoU $<$ 0.3. We investigated that the training using higher number proposals showing promising results. SS has a mAP of 0.55 under low-light followed by rainy condition 0.53; 0.52 for foggy and 0.51 for dust while using 2 k proposals and refinement strategy (I) which leading from refinement strategy (II). A same scenario has highlighted in other proposals numbers, i.e., 300 proposals and 1 k proposals. Till then we can also sense that these mAP values are not up to the promising detection accuracy. The reason might be the SS cannot merge up the super-pixels over smooth thermal frames as discussed in Fig. 1. Therefore, we investigated the single-stage-based approach, i.e., generation of pre-defined anchor boxes over feature cells [11, 15]. By default, we use three sets of three scales and three aspect ratios, i.e., Set_1—(2, 4, 8) scales and (0.5, 1, 2) aspect ratios, Set_2—(3, 6, 12) scales and (0.5, 1, 2) aspect ratios, Set_3—(4, 8, 16) scales and (0.5, 1, 2) aspect ratios. The mAPs are approximately equal in Set_1 and Set_2, and the mAPs are drops by a considerable margin of 1–3% in Set_3. The Set_3 generating anchor boxes perhaps larger than our ground truth bounding boxes. When we merged up these 3 sets into one, i.e., Set_4 is (2, 3, 4, 6, 8, 12, 16) scales and (0.5, 1, 2) aspect ratios, we are getting the highest mAP of 0.79 in both atmospheric conditions, i.e., low-light and rain, and mAP of 0.75 in rest two atmospheric conditions, i.e., dust and foggy. Therefore, the upcoming analysis in this study will be based on Set_4 anchor boxes.

Comparative Analysis of our AWRDNet model with other existing state-of-the-art models On the whole TU-VDN dataset, we compare AWRDNet against YOLOR [53], fast-RCNN [10], faster-RCNN [11], YOLO [13], YOLOv4

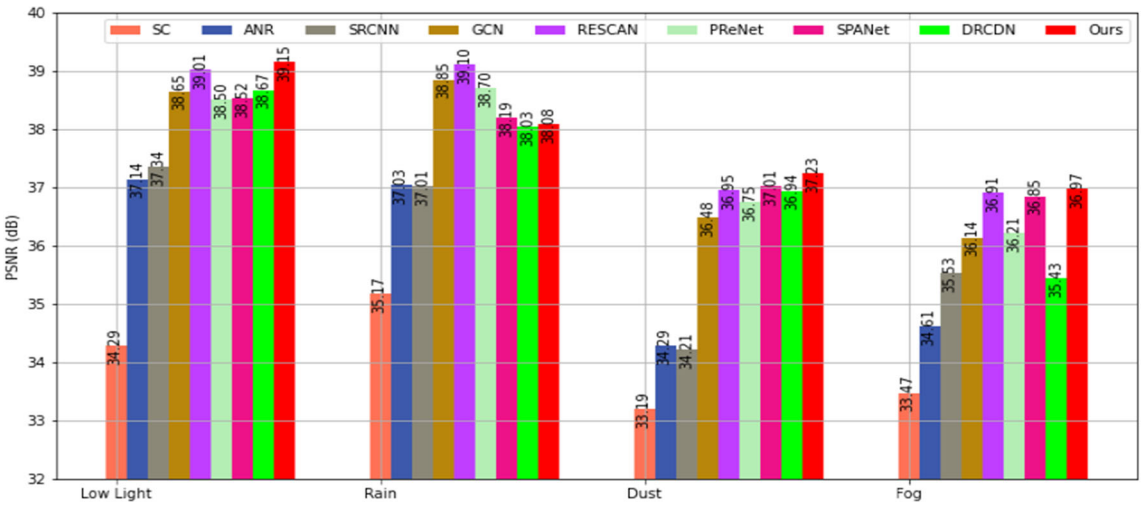
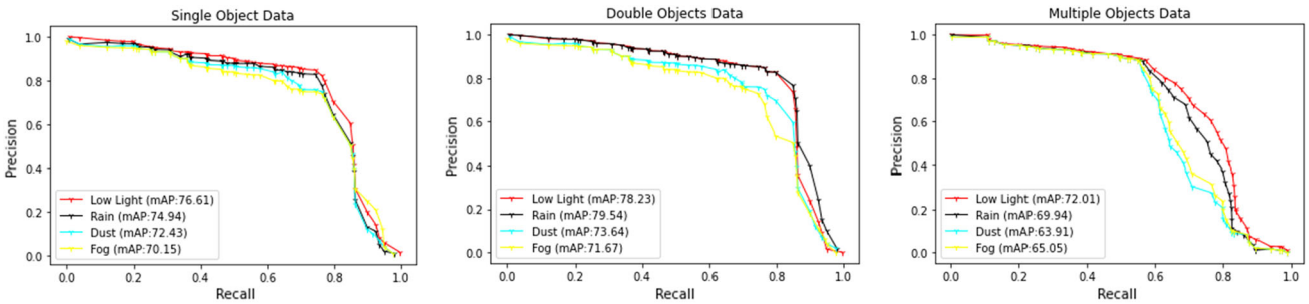


Fig. 11 Performance and comparative evaluation of restoration part of proposed model via PSNR. The state-of-the-art models are SC [41], ANR [42], SRCNN [43], GCN [44], RESCAN [45], PreNet [46], SPANet [47], and DRCDN [48]



(a) Single Object based Frames (b) Double Objects based Frames (c) Multi Objects based Frames

Fig. 12 Recall versus precision mean average precision (mAP) analysis on the TU-VDN dataset. During analysis, positive IoU threshold (pos_t = 0.3) and negative IoU threshold ($0.1 \leq$ neg_t < 0.3), NMS IoU threshold is 0.6 along with confidence score threshold is 0.6

[54], G-RCNN [55], and SSD [15]. The proposed model including SSD300, Fast and Faster RCNN are fine-tuned on the pre-trained VGG16 network where YOLO is based on GoogleNet and You Only Learn One Representation (YOLOR) base-net is YOLOv4. The YOLOv4 pre-trained with CSPDarknet53 and Granulated-RCNN (G-RCNN) is a family of FastRCNN which is pre-trained with AlexNet as base network. The AWRDNet detection is analyses via pre-defined anchor box approach as well as SS box approach, and SS producing very poor results in all our ablation cases. Whereas the rest of existing state-of-the-art models such as YOLO, YOLOv4, YOLOR, SSD which are analyzed via default anchor box strategies, faster-RCNN via RPN box, G-RCNN via bounding box, and fast-RCNN via SS strategy.

We designed 4 ablation cases to evaluate more closely as shown in Fig. 13. In case 1 (Fig. 13a), we kept NMS thresholds as IoU is 0.6 and confidence score is 0.6. Our network model realizes an approximately 0.63% deterioration mAP over the best-performing model, namely, YOLOR; although fast-RCNN and AWRDNet with SS yields the poorest results. In case 2 (Fig. 13b), the suppression thresholds set as IoU is 0.6 and confidence score is 0.3 giving maximum detection accuracy of approx. 79.23% than rest of ablation cases. Whereas YOLOR and YOLOv4 exhibits as second and third best performing models with $\approx$ 79% and $\approx$ 78%, respectively. For the two-stage approach models, the faster-RCNN showing promising results with $\approx$ 73% mAP. As usual, fast-RCNN and AWRDNet with SS yields the poorest outcomes. The IoU and confidence score thresholds of NMS are sets 0.3 and 0.6, respectively, during the testing phase in case 3 (Fig. 13c). The proposed model and state-of-the-art models exhibit satisfactory performances, whereas YOLOR, G-RCNN, and YOLOv4 becoming as second, third, and fourth satisfactory models than AWRDNet which has the highest mAP of 67.23%. Most of worst performances has analyzed when we set threshold values of IoU is 0.3 and score is 0.3 in case 4 (Fig. 13d). That might be because of greater number of redundant, unwanted pre-defined bounding boxes as threshold values are low and tried to

Table 2 Comparison in terms of region proposals, and pre-defined anchor box approaches in AWRDN model on the TU-VDN dataset

Stages	Methods	Frame and sub-sample size	Anchor aspect ratios	Anchor scales	#Proposals/ anchors	Proposals/anchors offset refinement strategies	mAP			
							Low light	Rain	Dust	Fog
Two-stage	AWRDNet + SS	Frames size: 640 × 480	–	–	300	(I)	0.46	0.45	0.43	0.43
						(II)	0.43	0.44	0.41	0.42
					1000	(I)	0.48	0.48	0.47	0.46
						(II)	0.45	0.46	0.44	0.44
					2000	(I)	**0.55**	**0.53**	**0.51**	**0.52**
						(II)	0.52	0.51	0.47	0.48
Single-stage	AWRDNet + Anchor	Frames/FM size: 640 × 480	{0.5, 1, 2}	Set_1 = {2, 4, 8}	3 × 3 × 40 × 30 = 10,800	(I)	0.77	0.76	0.75	0.75
						(II)	0.71	0.70	0.66	0.66
				Set_2 = {3, 6, 12}	3 × 3 × 40 × 30 = 10,800	(I)	0.76	0.76	0.74	0.73
						(II)	0.71	0.71	0.66	0.65
		Sub sample: 16		Set_3 = {4, 8, 16}	3 × 3 × 40 × 30 = 10,800	(I)	0.74	0.75	0.73	0.73
						(II)	0.67	0.68	0.65	0.64
		FM cells: 40 × 30		Set_4 = {2, 3, 4, 6, 8, 12, 16}	3 × 7 × 40 × 30 = 25,200	(I)	**0.79**	**0.79**	**0.75**	**0.75**
						(II)	0.73	0.73	0.70	0.71

Bold fonts indicating our proposed model best performances

(I) POS INDEX IF IoU $\geq$ 0.5, NEG INDEX IF 0.1 $\leq$ IoU $<$ 0.5; (II) POS INDEX IF IoU $\geq$ 0.3, NEG INDEX IF 0.1 $\leq$ IoU $<$ 0.3

match the number of resultant boxes with ground truth boxes.

To provide a better visual understanding of the detection results, typical bounding box-based detection results are shown in Fig. 14 under various atmospheric conditions.

6.2 Evaluation on the PASCAL VOC dataset

We comprehensively evaluate our proposed network model on the PASCAL VOC 2007 and 2012 detection benchmark datasets [29]. The results obtained using the PASCAL VOC dataset are presented in Table 3, i.e., the detection accuracy over several two-stage and single-stage object detector methods, and their corresponding extension version over the years. We used the initial learning rate of $1e^{-4}$ for the first 100 training epochs, then used the learning rate of $1e^{-3}$ for the next 50 epochs, and then $1e^{-2}$ for another 50 epochs.

Results on VOC 2007 Referring to Table 3, our AWRDNet achieves 82.6% mAP for train set "07"and 82.8% mAP for train set "07+12" surpassing all methods. Our method outperforms fast-RCNN by 15.7% (82.6 vs. 66.9) for "07" and 12.8% (82.8 vs. 70.0) for "07+12", and faster-RCNN further reduces mAP differences by 12.7% (82.6 vs. 69.9) for "07" and 9.6% (82.8 vs. 73.2) for "07+12". In contrast, the earlier regions-based family network, i.e., RCNN along with the bounding box proposals (BB) produce lesser results than AWRDNet by the differences of 16.6% (82.6 vs. 66.0) for "07" when the base network is OxfordNet and 24.1% (82.6 vs. 58.5) for "07" when the base network is TorontoNet. The SPP BB makes it 23.4% for "07" with base network ZF5. Almost closer difference produces with our model through HyperNet and HyperNet speedup version (SP), i.e., 6.3% (82.6 vs. 76.3) for "07" and 7.8% (82.6 vs. 74.8) for "07", respectively. The OHEM method makes it 12.7% (82.6 vs. 69.9) for "07" and 8.2% (82.8 vs. 74.6) for "07+12" with base network VGG16. All the mentioned existing methods are two-stage network built. The method D_SCNet-127 is only showing most promising performance even than our proposed model with 88.9% accuracy for "07" test set and 83.8% accuracy for "12" test set.

In single-stage based methods, YOLO with base network GoogleNet achieved 63.4% mAP and 66.4% mAP with VGG16 as base network for train set "07+12," and achieved most higher mAP, i.e., 78.6% to its next version YOLOv2 with DarkNet19 base network which is competitive to our proposed model. For 300 × 300 input size, SSD300 obtains 68% mAP for "07" train set and 74.3% mAP for "07+12" train set. For 512 × 512 input size, SSD512 obtains 71.6% detection accuracy for "07" train set and 76.8% detection accuracy for "07+12" train set. Trained with "07+12" set over base network ResNet-101, the SSD with 321 × 321 input size gets 77.1% mAP, and 80.6% mAP with 513 × 513 input size which almost similar accuracy with our proposed method having only 1.8% difference (82.8 vs. 80.6). The deconvolution version

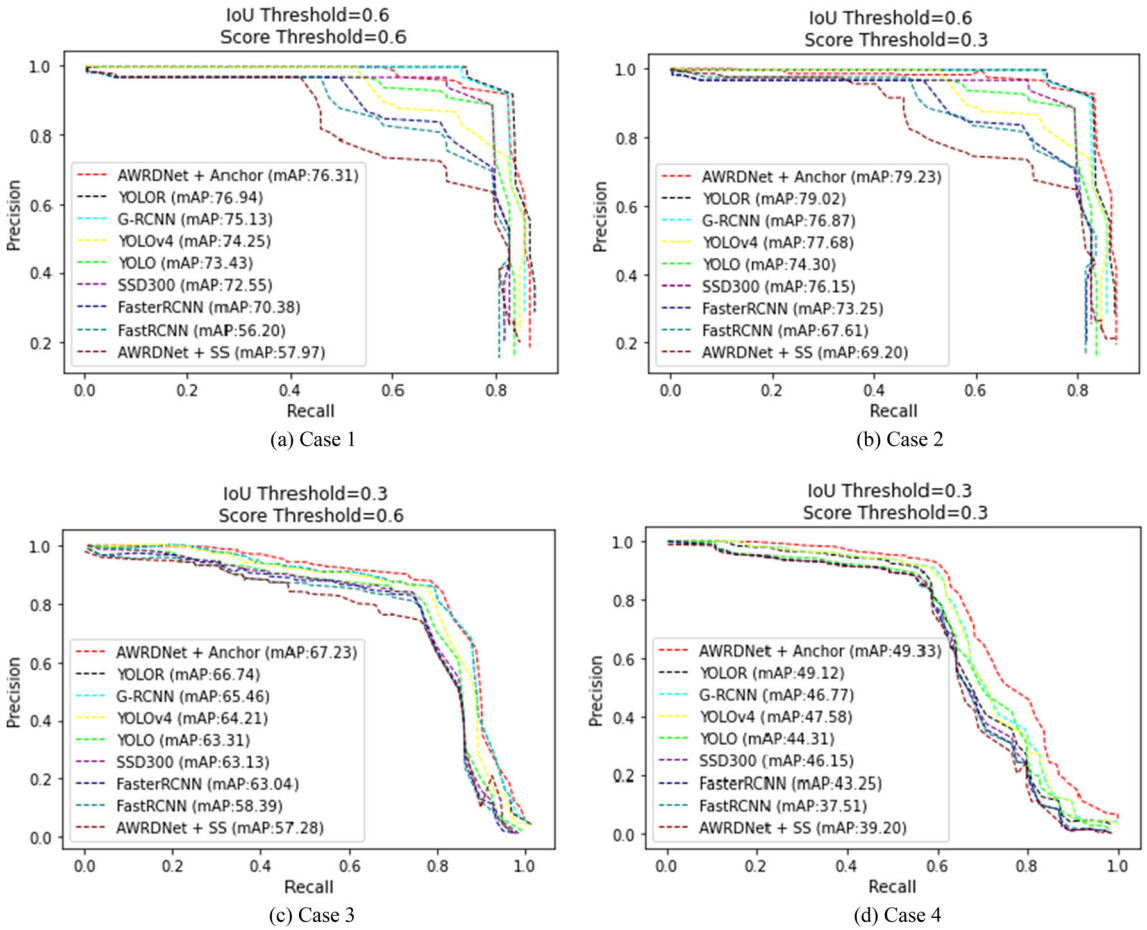


Fig. 13 Analysis of CNN models via recall versus precision graph in the whole TU-VDN dataset via NMS thresholds

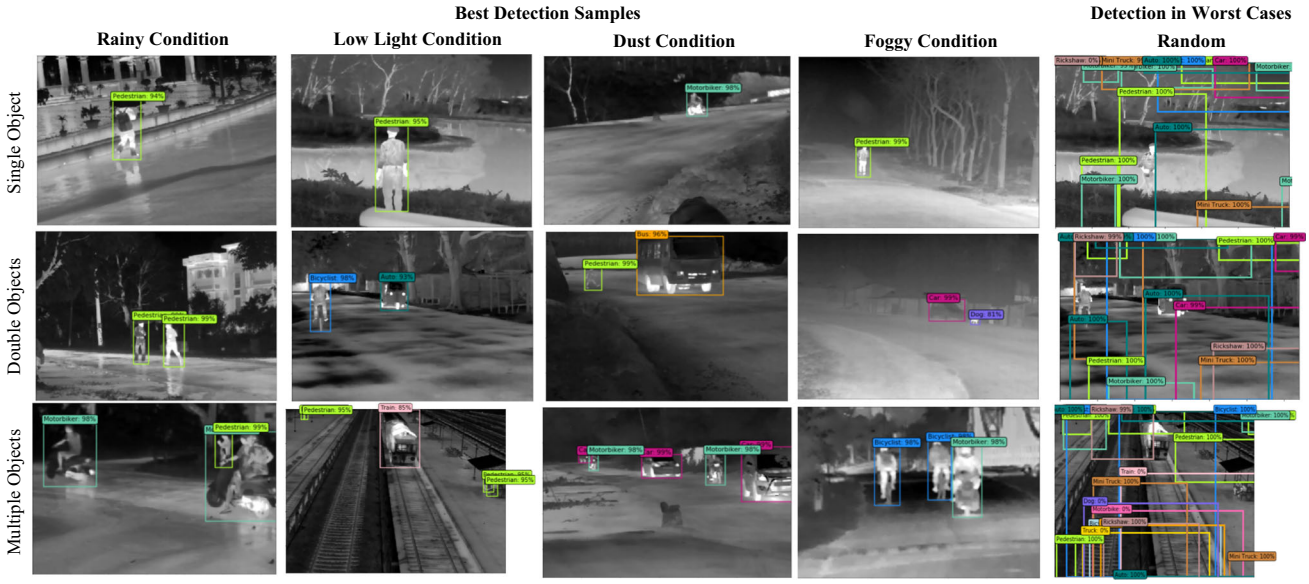


Fig. 14 Typical detection results of various atmospheric conditions in our created night time dataset. Column (1) shows detection results under rainy condition, column (2) shows detection results under low light condition, column (3) shows detection results under dusty condition, column (4) shows detection results under foggy condition, column (5) shows a randomly selected some worst results

of SSD on 321 input size, i.e., DSSD321 achieved competitive accuracy of 78.6% on "07+12" train set where ResNet-101 was used as base network. Input size of 513 × 513 on train set "07+12" gets successful detection accuracy of 81.5% which very close to our proposed method accuracy, i.e., 82.8% mAP. In case of deeply supervised method—DSOS and fully convolutional framework—RON achieved competitive detection accuracy against AWRDNet of 77.7% and 77.6%, respectively. A recent work, namely, Deep RegionletsA and Deep RegionletsP versions achieved good amount of accuracy of 80.1% and 80.3%, respectively, for "07+12" set.

Results on VOC 2012 In case of VOC 2012 dataset, our AWRDNet achieves 81.8% mAP for train set "07+12" and 81.4% mAP for train set "07++12" surpassing all methods except D_SCNet-127 (83.8%). When compared to fast-RCNN, our method outperforms it by 13.0% (81.4 vs. 68.4) for "07++12," and faster-RCNN further reduces mAP differences by 11.0% (81.4 vs. 70.4). Our model makes analogous differences with HyperNet and HyperNet speedup version (SP), i.e., 10.4% (81.8 vs. 71.4) and 10.5% (81.8 vs. 71.3) for "07+12", respectively. The OHEM method also makes it 9.5% (81.4 vs. 71.9) for "07++12" with base network VGG16.

In the single-stage based methods, YOLO with base network GoogleNet achieved 57.9% mAP for train set "07+12," and achieved higher mAP, i.e., 73.4% to its next version YOLOv2 with DarkNet19 base network for "07++12". For 300 × 300 input size, SSD300 obtains 72.4% mAP for "07++12" train set and 74.9% mAP for 512 × 512 input size—SSD512. Trained with base network ResNet-101, the SSD with 321 × 321 input size gets 75.4% mAP, and 79.4% mAP with 513 × 513 input size for "07++12" train set. The deconvolution version of SSD on 321 input size, i.e., DSSD321 achieved accuracy of 76.3% on "07++12" train set where ResNet-101 was used as base network. Input size of 513 × 513 on train set "07++12" gets successful detection accuracy of 80% which almost similar accuracy with our proposed method having only 1.4% difference (81.4 vs. 80.0). In case of deeply supervised method—DSOS and fully convolutional framework—RON achieved comparable detection accuracy against AWRDNet of 76.3% and 75.4%, respectively, for "07++12" train set.

6.3 Comparative assessment on the ZUT dataset

A very few datasets have been published for detection of objects in adverse weather conditions over the decades. Since our current study is on thermal dataset, we have chosen another publicly available thermal dataset, namely, ZUT [57] as having the extensive variety of captured data in the four European Union countries such as Denmark, Germany, Poland, and Lithuania. The dataset captured during severe weather conditions like drizzle, rain, cloudy, frost, fog, and clear sky. ZUT dataset that contains 122 k annotations collected during the drizzle or the rain, only 752 annotations were perceived during the clear sky, and the remaining annotations were collected during frosty and cloudy conditions [57]. The ZUT dataset is publicly accessible at Github and IEEE Dataport.

The comparative assessments results are shown in Table 4. The methods are used same as Sect. 6.1.3, i.e., Fast-RCNN, Faster-RCNN, G-RCNN, SSD, YOLO, YOLOv4, YOLOR, and ours (AWRDNet). From the severe weather conditions point of view, the YOLOR model shows as best performing with 0.69% F-score for drizzle weather condition where YOLOv4 and AWRDNet as second best. In case of rainy condition, the AWRDNet model shows promising with 0.70% whereas 0.68% for both YOLOv4 and YOLOR. For cloudy condition, the G-RCNN, YOLOv4, and YOLOR are with equal metric values 0.70%, although rest of models also shows competitive performances. In both the frosty and fog conditions, the models decreases in their performances with highest 0.69% YOLOR and lowest 0.59% Fast-RCNN for frosty condition, and with highest 0.69% AWRDNet and lowest 0.57% Faster-RCNN for foggy condition. In case of clear sky, the models improves a lot where AWRDNet shows 0.75% metric value with most promising and 0.73% as second best metric values for both YOLOv4 and YOLOR.

7 Complexity analysis

In this section, we present the estimation of the parameters and speed of the proposed approach.

Number of parameters FC layers bring more parameters than YOLO and SSD. But, we estimated the number of parameters for our proposed architecture convolutional blocks, as presented in Table 5. Similarly, YOLO [13] and SSD [15] are also calculated for comparison purpose, where YOLO consists of 24 convolutional layers, which gives about "$X + 80.73$" million parameters, and SSD has 10 convolutional layers, which gives "$X + 10.3$" million parameters. The "X" is the number of parameters from the base network, which includes transfer learning parameters. Furthermore, our proposed network provide lesser number of parameters, such as "$X + 1.3$" million for $N = 35$ and "$X + 0.9$" million for $N = 30$ and so on.

Runtime speed We evaluate the runtime of our proposed approach and compare with other single-stage object detectors using the PyTorch framework. The time is

Table 3 Results on PASCAL VOC 2007 and 2012 test dataset

Stage	Study	Method	Base network	Train data	#Proposals/ anchors	mAP (%)	
						VOC 2007 test set	VOC 2012 test set
Two-stage	Girshick et al. [10]	FastRCNN	VGG16	07	2000	66.9	–
	Girshick et al. [10]	FastRCNN	VGG16	07+12	2000	70.0	–
	Girshick et al. [10]	FastRCNN	VGG16	12	2000	–	65.7
	Girshick et al. [10]	FastRCNN	VGG16	07++12	2000	–	68.4
	Ren et al. [11]	FasterRCNN	VGG16	07	300	69.9	–
	Ren et al. [11]	FasterRCNN	VGG16	07+12	300	73.2	–
	Ren et al. [11]	FasterRCNN	VGG16	12	300	–	67.0
	Ren et al. [11]	FasterRCNN	VGG16	07++12	300	–	70.4
	Girshick et al. [9]	R-CNN BB	OxfordNet	07	2000	66.0	–
	Girshick et al. [9]	R-CNN BB	TorontoNet	07	2000	58.5	–
	He et al. [20]	SPP BB	ZF5	07	–	59.2	–
	Kong et al. [21]	HyperNet	VGG16	07	100	76.3	–
	Kong et al. [21]	HyperNet SP	VGG16	07	100	74.8	–
	Kong et al. [21]	HyperNet	VGG16	07+12	100	–	71.4
	Kong et al. [21]	HyperNet SP	VGG16	07+12	100	–	71.3
	Shrivastava et al. [22]	OHEM	VGG16	07	300	69.9	–
	Shrivastava et al. [22]	OHEM	VGG16	07+12	300	74.6	–
	Shrivastava et al. [22]	OHEM	VGG16	12	300	–	69.8
	Shrivastava et al. [22]	OHEM	VGG16	07++12	300	–	71.9
	Kim et al. [49]	BBCNet	FasterRCNN	07	–	73.2	–
	Kim et al. [49]	BBCNet	FasterRCNN	07+12	–	74.9	–
	Quan et al. [52]	D_SCNet-127 R-CNN	R-CNN	07	–	88.9	83.8
Single-stage	Redmon et al. [13]	YOLO	GoogleNet	07+12	98	63.4	–
	Redmon et al. [13]	YOLO	VGG16	07+12	98	66.4	–
	Redmon et al. [13]	YOLO	GoogleNet	07+12	98	–	57.9
	Redmon et al. [14]	YOLOv2	DarkNet19	07+12	1445	78.6	–
	Redmon et al. [14]	YOLOv2	DarkNet19	07++12	1445	–	73.4
	Liu et al. [15]	SSD300	VGG16	07	8732	68.0	–
	Liu et al. [15]	SSD300	VGG16	07+12	8732	74.3	–
	Liu et al. [15]	SSD512	VGG16	07	8732	71.6	–
	Liu et al. [15]	SSD512	VGG16	07+12	8732	76.8	–
	Liu et al. [15]	SSD300	VGG16	07++12	8732	–	72.4
	Liu et al. [15]	SSD512	VGG16	07++12	8732	–	74.9
	Fu et al. [18]	SSD321	ResNet101	07+12	17,080	77.1	–
	Fu et al. [18]	SSD513	ResNet101	07+12	43,688	80.6	–
	Fu et al. [18]	SSD321	ResNet101	07++12	17,080	–	75.4
	Fu et al. [18]	SSD513	ResNet101	07++12	43,688	–	79.4
	Fu et al. [18]	DSSD321	ResNet101	07+12	17,080	78.6	–
	Fu et al. [18]	DSSD513	ResNet101	07+12	43,688	81.5	–
	Fu et al. [18]	DSSD321	ResNet101	07++12	17,080	–	76.3
	Fu et al. [18]	DSSD513	ResNet101	07++12	43,688	–	80.0
	Shen et al. [16]	DSOD	DenseNet	07+12	8732	77.7	–

Table 3 (continued)

Stage	Study	Method	Base network	Train data	#Proposals/ anchors	mAP (%)	
						VOC 2007 test set	VOC 2012 test set
	Shen et al. [16]	DSOD	DenseNet	07++12	8732	–	76.3
	Kong et al. [36]	RON384++	VGG16	07+12	30,600	77.6	–
	Kong et al. [36]	RON384++	VGG16	07++12	30,600	–	75.4
	Xu et al. [50]	Deep RegionletsA	VGG16	07	–	73.8	–
	Xu et al. [50]	Deep RegionletsA	VGG16	07+12	–	80.1	–
	Xu et al. [50]	Deep RegionletsP	VGG16	07	–	73.9	–
	Xu et al. [50]	Deep RegionletsP	VGG16	07+12	–	80.3	–
	Ma et al. [51]	MDFN-11	VGG16	07+12	–	79.3	–
	Ma et al. [51]	MDFN-12	VGG16	07+12	–	78.3	–
	Ours	**AWRDNet**	**VGG16**	**07**	*	**82.6**	**–**
		AWRDNet	**VGG16**	**07+12**	*	**82.8**	**81.8**
		AWRDNet	**VGG16**	**07++12**	*	**–**	**81.4**

Bold fonts indicating our proposed model best performances

"07": VOC 2007 TrainVal, "12": VOC 2012 TrainVal, "07+12": union set of VOC 2007 TrainVal and VOC 2012 TrainVal, "07++12": union set of VOC 2007 TrainVal+Test and VOC 2012 TrainVal

*Number of anchors will be decided after refinement

reported on a workstation server with Intel Xeon CPU, 48 GM RAM, NVIDIA TITAN XP GPU, CUDA 8.0 implementation excluding data pre-processing. On an average, YOLO [13] takes 22.22 ms and SSD [15] takes 16.95 ms per frame, while our approach takes 20.83 ms. Our approach has lesser number of parameters in the convolutional layers than SSD and YOLO; however, its time consumption is more than that of SSD and less than that of YOLO. This is because the deconvolutional operations provide high-resolution FM and correspondingly increase the number of parameters in fully connected layers.

Table 4 Performance evaluation on ZUT dataset using F1-score metric

Methods	Severe weather conditions					
	Drizzle	Rain	Cloudy	Frost	Fog	Clear sky
Fast-RCNN	*0.61*	*0.63*	*0.68*	*0.59*	0.58	*0.70*
Faster-RCNN	0.63	0.68	*0.68*	0.61	*0.57*	0.71
G-RCNN	0.67	0.68	**0.70**	0.66	0.63	0.72
SSD	0.65	0.67	0.69	0.64	0.64	0.71
YOLO	0.66	0.67	0.69	0.65	0.66	0.72
YOLOv4	0.67	0.68	**0.70**	0.64	0.67	0.73
YOLOR	**0.69**	0.68	**0.70**	**0.69**	0.68	0.73
AWRDNet	0.67	**0.70**	0.69	0.68	**0.69**	**0.75**

Italic fonts indicates the overall worst and bold fonts indicates the overall best performances

8 Conclusion

In this study, we recommended a single-stage CNN architecture, namely, AWRDNet, for restoration cum object detection in real-time adverse atmospheric scenes. In this correspondence, we created bounding box ground-truth annotations on our TU-VDN dataset and data augmentations for detecting objects. We summarize the outcomes of this proposed architecture as follows. (a) A feed-forward deeper convolutional layer produces better quality of restoration images, wherein receptive field plays an important role in analysing local features over degraded scenes. (b) Another key feature of the proposed model is the clipping of 21 pre-defined multi-scale anchor boxes per cell to a restorated de-convolutional FM, which allows us to efficiently reduce time-consumption. (c) In terms of detection enactment, the results of the comparative experiments on the TU-VDN dataset demonstrated the optimal performance of our proposed model. It also revealed that the performance accuracy in low-light or rainy conditions is higher than that in dusty or foggy conditions. (d) The analysis on the PASCAL VOC dataset and ZUT thermal dataset demonstrated the comparative assessment of the proposed approach than other recent two-stage and single-stage state-of-the-art networks. At the last, the complexity of the proposed architecture has drawn with total number of parameters $X + 29542N$.

To access the dataset, kindly send the user agreement form from http://www.mkbhowmik.in/tuvdn.aspx.

Table 5 Estimation of number of convolutional layer-based parameters for our proposed architecture

Layer name	Tensor size	Weights ($W_C = K^2 \times C \times N$)	Biases ($B_C = N$)	Parameters ($P_C = W_C + B_C$)
Input image	640 × 480 × 3	0	0	0
Base net	–	–	–	X
Block_K for $K = 1, \ldots, N$				
ConvK_1	224 × 224 × 64	$3^2 \times 64 \times 64$	64	36,928
BN	224 × 224 × 64	0	0	0
ReLU	224 × 224 × 64	0	0	0
Dropout ($p = 0.8$)	224 × 224 × 64	0	0	$36928 \times (1 - p) = 7386$
For N convolutional blocks, the number of parameters will be				$= (36928 - 7386)N = 29542N$
DC				
AM	320 × 240 × 64	0	0	0*
MU	640 × 480 × 64	0	0	0*
Total number of parameters				$= X + 29542N$

K—size (width) of kernels used in the Conv layer, *N*—number of kernels, *C*—number of channels of the input image, *BN* batch normalization, *DC* De-convolution, *AM* adaptive MaxPooling, *MU* MaxUnpooling

*No parameters are associated with the AdaptiveMaxPooling and MaxUnpooling, pool size, unpool size, stride, and padding are hyperparameters

Acknowledgements This study was conducted in the Computer Vision Laboratory of Computer Science and Engineering Department of Tripura University (A Central University), Tripura, Suryamani-nagar-799022.

Funding There is no funding for this research.

Data availability Data will be made available on reasonable request.

Declarations

Conflict of interest The author declares no potential conflict of interest with respect to the authorship and/or publication of this article.

References

1. Narasimhan SG, Nayar SK (2002) Vision and the atmosphere. Int J Comput Vis 48:233–254
2. Chen CC (1975) Attenuation of electromagnetic radiation by haze, fog, clouds, and rain. A report prepared by United States Air Force Project Rand
3. Dong C et al (2015) Image super resolution using deep convolutional networks. In: TPAMI
4. Kim J et al (2015) Accurate image super-resolution using very deep convolutional networks. In: CVPR, pp 1646–1654.
5. Rumar K (2003) Infrared night vision systems and driver needs. SAE, Warrendale
6. Singha A, Bhowmik MK (2019) TU-VDN: Tripura University Video Dataset at Night Time in degraded atmospheric outdoor conditions for moving object detection. In: Proceedings of 26th IEEE international conference on image processing (ICIP), pp 2936–2940
7. Singha A, Bhowmik MK (2019) Salient features for moving object detection in adverse weather conditions during night time. IEEE Trans Circuits Syst Video Technol 6:66
8. Zhao Z et al (2018) Object detection with deep learning: a review. IEEE Trans Neural Netw Learn Syst. https://doi.org/10.1109/TNNLS.2018.2876865
9. Girshick R et al (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference computing vision pattern recognition, Columbus, USA, pp 580–587
10. Girshick R (2015) Fast R-CNN. In: Proceedings of the IEEE international conference computing vision, Santiago, Chile, pp 1440–1448
11. Ren S et al (2015) Faster R-CNN: towards realtime object detection with region proposal networks. In: Proceedings of the advances neural information processing systems, Montreal, Canada, pp 91–99
12. Uijlings JR et al (2013) Selective search for object recognition. Int J Comput Vis 6:66
13. Redmon J et al (2016) You only look once: unified, real-time object detection. In: Proceedings of the IEEE conference computing vision pattern recognition, Las Vegas, USA, pp 779–788
14. Redmon J, Farhadi A (2016) YOLO9000: better, faster, stronger, arXiv e-prints, arXiv:1612.08242v1 [cs.CV]
15. Liu W et al (2016) SSD: single shot multibox detector. In: Proceedings of the European conference on computing vision, Amsterdam, Netherlands, pp 21–37
16. Shen Z et al (2017) DSOD: learning deeply supervised object detectors from scratch. In: Proceedings of the IEEE international conference on computing vision, Venice, Italy, pp 1919–1927
17. Cai Z et al (2016) A unified multiscale deep convolutional neural network for fast object detection. In: ECCV
18. Fu C et al (2017) DSSD: deconvolutional single shot detector, arXiv:1701.06659
19. Hoffman J et al (2014) From large-scale object classifiers to large-scale object detectors: an adaptation approach. In: NIPS
20. He K et al (2014) Spatial pyramid pooling in deep convolutional networks for visual recognition. In: ECCV
21. Kong T et al (2016) HyperNet: towards accurate region proposal generation and joint object detection. In: CVPR, pp 845–853

22. Abhinav S et al (2016) Training region-based object detectors with online hard example mining. In: CVPR, pp 761–769
23. Szegedy C et al (2013) Deep neural networks for object detection. In: NIPS
24. Sermanet P et al (2014) Overfeat: integrated recognition, localization and detection using convolutional networks. In: International conference on learning representations (ICLR)
25. Erhan D et al (2014) Scalable object detection using deep neural networks. In: IEEE conference on computer vision and pattern recognition (CVPR)
26. Szegedy C et al (2015) Scalable, high-quality object detection, arXiv:1412.1441 (v1)
27. Zitnick CL, Dollár P (2014) Edge boxes: locating object proposals from edges. In: ECCV
28. Hosang J et al (2015) What makes for effective detection proposals? arXiv e-prints, arXiv:1502.05082v1 [cs.CV]
29. Everingham M et al (2010) The pascal visual object classes (voc) challenge. Int J Comput Vis 88(2):303–338
30. Lin TY et al (2014) Microsoft coco: common objects in context. In: Proceedings of the European conference on computing vision, Zurich, Switzerland, pp 740–755
31. Russakovsky O et al (2015) ImageNet large scale visual recognition challenge. Int J Comput Vis 115(3):211–252
32. Schwarz MW, Cowan WB, Beatty JC (1987) An experimental comparison of RGB, YIQ, LAB, HSV, and opponent color models. ACM Trans Graph 6(2):123–158
33. Narasimhan SG, Nayar SK (2003) Contrast restoration of weather degraded images. IEEE Trans Pattern Anal Mach Intell 25(6):713–724
34. Felzenszwalb PF et al (2010) Object detection with discriminatively trained part based models. IEEE Trans Pattern Anal Mach Intell 6:66
35. Timofte R et al (2013) Anchored neighbourhood regression for fast example-based super-resolution. In: ICCV
36. Kong T et al (2017) RON: reverse connection with objectness prior networks for object detection. In: CVPR, pp 5936–5944
37. Li X, Ye M, Liu Y, Zhu C (2017) Adaptive deep convolutional neural networks for scene-specific object detection. IEEE Trans Circuits Syst Video Technol 29(9):2538–2551
38. Chen X, Li H, Wu Q, Ngan KN, Xu L (2020) High-quality R-CNN object detection using multi-path detection calibration network. IEEE Trans Circuits Syst Video Technol 66:1
39. Jie Z, Lu WF, Sakhavi S, Wei Y, Tay EHF, Yan S (2016) Object proposal generation with fully convolutional networks. IEEE Trans Circuits Syst Video Technol 28(1):62–75
40. Chen M, Zhang J, He S, Yang Q, Li Q, Yang M-H (2017) Interactive hierarchical object proposals. IEEE Trans Circuits Syst Video Technol 29(9):2552–2566
41. Yang J, Wright J, Huang T, Ma Y (2008) Image super-resolution as sparse representation of raw image patches. In: Proceedings of the CVPR, pp 1–8
42. Timofte R, De Smet V, Van Gool L (2013) Anchored neighborhood regression for fast example-based super-resolution. In: Proceedings of the ICCV, pp 1920–1927
43. Dong C, Loy CC, He K, Tang X (2014) Learning a deep convolutional network for image super-resolution. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T (eds) Computer vision—ECCV, vol 8692
44. Fu X, Qi Q, Zha ZJ et al (2021) Successive graph convolutional network for image de-raining. Int J Comput Vis 129:1691–1711
45. Li X, Wu J, Lin Z, Liu H, Zha H (2018) Recurrent squeezeand-excitation context aggregation net for single image deraining. In: Proceedings of the ECCV
46. Ren D, Zuo W, Hu Q, Zhu P, Meng D (2019) Progressive image deraining networks: a better and simpler baseline. In: Proceedings of the CVPR
47. Wang T, Yang X, Xu K, Chen S, Zhang Q, Lau RW (2019) Spatial attentive single-image deraining with a high quality real rain dataset. In: Proceedings of the CVPR
48. Zhang S, He F (2020) DRCDN: learning deep residual convolutional dehazing networks. Vis Comput 36:1797–1808
49. Kim JU, Kwon J, Kim HG, Ro YM (2019) BBC net: bounding-box critic network for occlusion-robust object detection. IEEE Trans Circuits Syst Video Technol 30:1037–1050
50. Xu H, Lv X, Wang X, Ren Z, Chellappa R (2021) Deep regionlets: blended representation and deep learning for generic object detection. IEEE Trans Pattern Anal Mach Intell 43:1914–1927
51. Ma W, Wu Y, Cen F, Wang G (2020) MDFN: multi-scale deep feature learning network for object detection. Pattern Recognit 100:107–149
52. Quan Y, Li Z, Chen S et al (2021) Joint deep separable convolution network and border regression reinforcement for object detection. Neural Comput Appl 33:4299–4314
53. Wang C-Y, Yeh I-H, Liao H-YM (2021) You only learn one representation: unified network for multiple tasks. arXiv:2105.04206v1 [cs.CV]
54. Bochkovskiy A, Wang C-Y, Liao H (2020) YOLOv4: optimal speed and accuracy of object detection
55. Pramanik A, Pal SK, Maiti J, Mitra P (2022) Granulated RCNN and multi-class deep SORT for multi-object detection and tracking. IEEE Trans Emerg Top Comput Intell 6:171–181. https://doi.org/10.1109/TETCI.2020.3041019
56. Yu F, Koltum V (2016) Multi-scale context aggregation by dilated convolutions, arXiv:1511.07122v3 [cs.CV]
57. Tumas P, Nowosielski A, Serackis A (2020) Pedestrian detection in severe weather conditions. IEEE Access 8:62775–62784