Applied Intelligence
https://doi.org/10.1007/s10489-022-03245-5

# Effects of haze and dehazing on deep learning-based vision models

Haseeb Hassan[1,2,3] · Pranshu Mishra[4] · Muhammad Ahmad[5] · Ali Kashif Bashir[6] · Bingding Huang[1,3] · Bin Luo[7]

Accepted: 14 January 2022


**Abstract**
Most deep-learning-based vision models are trained and tested on clear images, avoiding noisy, or hazy, images. However, these models may encounter degraded images. So, it is important to recover and enhance them using a dehazing process. Dehazing usually serves as a preprocessing step for low-, medium-, and high-level vision tasks. Therefore, this article empirically studies the impact of haze and dehazing on high-level vision tasks and considers the degree to which dehazing algorithms can improve a vision model's performance. For this purpose, we created two synthetic hazy datasets and trained several detection and classification models on both clear and hazy images. We found that haze and fog can easily affect a vision model's performance and observed that using dehazing directly as a preprocessing step for high-level vision tasks did not substantially improve vision model's performance but also renders performance unreliable and unpredictable. Therefore, when developing deep vision models, the research community should maintain aspects of bad weather conditions, such as haze, mist, fog, and rain, to avoid the failure of their proposed outdoor vision models.

**Keywords** Image dehazing · Image classification · Image detection · High-level · Vision models · Deep-learning-based vision models

## 1 Introduction

Deep-learning-based vision models depend upon certain filter outputs to achieve classification and recognition tasks. Mostly, these models are trained and validated using clear images [1]. However, in practical outdoor vision applications, a model is likely to encounter degraded images. For instance, due to haze, images captured in bad weather conditions often suffer from poor visibility and low contrast [2]. Haze impairs an image's quality and complicates the subsequent processing and analysis [3]. Furthermore, haze degrades an outdoor image's visibility and impacts numerous high-level vision applications, including surveillance, autonomous vehicles, aerial robots, and intelligent infrastructure [4]. This is because it is difficult to detect and classify objects and their structural information in hazy conditions. Thus, dehazing signifies the only solution, enabling a model to restore a haze-free image from a hazy image. The dehazing procedure has received substantial attention from the computer vision research community in recent years [5, 6], with the literature consistently acknowledging that properly dehazed hazy images will become comprehensible and capable of providing supplementary assistance for vision tasks. This suggests that image dehazing is crucial

✉ Bingding Huang
huangbingding@sztu.edu.cn

✉ Bin Luo
luobin@ahu.edu.cn

1 College of Applied Sciences, Shenzhen University, Shenzhen 518060, China

2 Guangdong Key Laboratory for Biomedical Measurements and Ultrasound Imaging, School of Biomedical Engineering, Shenzhen University Health Science Center, Shenzhen 518060, China

3 College of Big Data and Internet, Shenzhen Technology University, Shenzhen 518188, China

4 Department of Computer Science & Engineering, SJB Institute of Technology, Bangalore, India

5 Department of Computer Science, National University of Computer and Emerging Sciences, Islamabad, Chiniot-Faisalabad Campus, Chiniot 35400, Pakistan

6 Department of Computing and Mathematics, Manchester Metropolitan University, Manchester, UK

7 School of Computer Science and Technology, Anhui University, Hefei, China

Published online: 22 March 2022

to consumer photography and computer vision applications [4, 6].

This paper analyzes the effects of haze and dehazing on deep-learning-based vision models. Thus far, two works [7, 8] have probed these effects in the context of classification tasks. Meanwhile, to the best of our knowledge, no study has fully addressed the impact of haze and dehazing on computer vision-based detection tasks. This research produces new insights regarding haze and the dehazing problem in the context of high-level vision tasks, including the degree to which dehazing is suitable for high-level vision tasks. Although it is understood that dehazing does not substantially aid classification tasks [8]. In contrast, the detection tasks are not investigated in terms of haze and dehazing effects. Thus, it is also imperative to investigate the detection tasks in terms of haze and dehazing effects.

## 2 Related work

Haze degrades image quality, and dehazing upgrades image quality, ultimately aiming to restore a clear image that is not only more visually appealing but also assists high-level vision applications. Various dehazing approaches have been proposed, including a non-local dehazing network [9] and an end-to-end deep residual convolutional dehazing network [10]. High-level vision systems have received significant attention in recent years, especially the practices of classification and detection. The handcrafted features [11] were initially used for image classification. Later on, the evolution of deep learning led to significant research considering large-scale labeled datasets, including ImageNet [12], PASCAL VOCs [13], and Microsoft Common Objects in Context (COCO) [14]. For example, research into classification using deep convolutional neural networks (CNNs) [15], such as VGGNet [16], Inception [17], residual learning [18], and Mobilenets [19], have demonstrated outstanding image classification performance in the context of clear images. Similarly, researchers have achieved better image classification by introducing a cross-convolutional-layer pooling procedure for image classification [20] and pooling the CNN with recurrent neural networks (RNN) [21]. Meanwhile, Durand et al. [22] projected a learning-based method that efficiently executed image classification, object localization, and segmentation. Fei et al. [23] anticipated a "Residual Attention Network" classification model by considering an attention mechanism incorporating the feed-forward network in an end-to-end style.

Besides classification tasks, other high-level vision tasks, including object detection, have also received significant attention. For example, a region-based convolutional network detector (R-CNN) using a deep ConvNet to classify object proposals has demonstrated excellent object detection [24]. However, the R-CNN has notable drawbacks, notably speed, which prompted the development of the spatial pyramid pooling network [25], which computes a convolutional feature map for the entire input image and then classifies each object proposal using a feature vector extracted from the shared feature map. Inspired by that, a Fast R-CNN method [26] was introduced to classify object proposals using deep convolutional networks more efficiently, which meant modifying several settings to improve training and testing in terms of detection accuracy. Shaoqing et al. [27] subsequently proposed a Faster R-CNN technique (FRCNN) based on a region proposal network (RPN) that shared full-image convolutional features and a detection network and enabled cost-free region proposals. Jiahui et al. [28] proposed a detection technique involving the Intersection over Union loss function predicting the bounding box that efficiently localized image objects. Several other state-of-the-art detection models have also been introduced for object detection and demonstrated good performance [29–32].

The CNN-based vision architectures perform well on clear images. Though, in some circumstances, good-quality images are not attainable. Several techniques have explored classification accuracy in the context of such cases. For instance, for face recognition in low-resolution images, one study suggested a kernel embedding method [33]. Zou et al. [34] adopted discriminative constraints to restore high-quality images for the recognition task. Elsewhere Basu et al. [35] synthesized the MNIST dataset [36] by generating noisy handwritten images and proposed a model based on sparse representation to classify those noisy generated MNIST digits. In [37] a noisy face benchmark was developed for recognition purposes. Meanwhile, Prasun et al. [38] studied the effects of degradation on deep neural network architectures, specifically considering motion blur, Gaussian, and salt-and-pepper noises.

It is generally understood that image degradation affects the performance of vision systems. This prompted Zhang et al. [39] to simultaneously incorporate feature enhancement and recognition. Likewise, Steven et al. [40] intend an end-to-end network that could jointly conduct denoising and deblurring processes for classification tasks. The famous AOD-Net technique embeds a trained model into the FRCNN to attain superior dehazing outcomes for the detection task [41]. In recent past, Pei et al. [8] investigated dehazing and its impact on classification problems and exposed the negligible effects of dehazing. Their proposed investigative study also provides us the intuition to investigate the haze and dehazing impact on detection tasks.

The rest of the article is organized as follows. Section 3 reports on experiments conducted in relation to classification tasks. Section 4 probes haze and dehazing in the context of detection tasks. Section 5 summarizes all of the experiments,

discusses the findings and insights, and remarks on future perspectives for the topic.

## 3 Haze and dehazing analysis for classification tasks

For classification tasks, we considered state-of-art deep convolution classification networks [16], namely, VGG-16, VGG-19, and deep residual learning (ResNet) [18]. First, these classifiers were trained and tested on clear images. Then, we tested the trained classifiers on hazy images. Next, we dehazed the hazy images and tested the trained classifiers on the dehazed images. Finally, we compared the trained model's performances to analyze the effects of dehazing.

### 3.1 Classification dataset and adding synthetic haze

Performing haze and dehazing analyses requires clear and hazy image sets. For this purpose, we selected the Caltech-256 object category dataset [42], which is divided into training (92%) and test (8%) sets. We further divided the test data (8%) into two subsets: clear test data (5%) and synthetic hazy test data (3%). We used a foggy haze simulator to generate synthetic hazy test data [43]. The synthetic hazy test data (3%) includes low, medium, and high-level hazy images, which could be used in our classification and detection analysis. Figure 1 shows some of the sample images from our synthetic datasets, whereas level 1, level 2, and level 3 hazy represents low, medium, and high-level hazy images.

#### 3.1.1 Experimental setup and results

This section introduces the three selected image classification architectures, and explains their training procedures. These trained models are then tested on clear test data (5%) and synthetic hazy test data (3%). We subsequently introduce eight state-of-the-art image-dehazing approaches and utilize them to dehaze synthetic hazy test data (3%). Finally, we evaluate the three trained classification frameworks on the dehazed images.

1. VGG-16 is a CNN architecture and is considered an excellent vision model, focusing on the convolution layers of a $3\times3$ filter with stride one instead of many hyper-parameters. It uses the same padding and MaxPool layer of $2\times2$ filter with stride 2 and follows the convolution and max pool layer arrangement reliably throughout the architecture. It ultimately includes two fully connected layers followed by a SoftMax. This network is vast and features approximately 138 million parameters. The pre-trained network can classify images into 1000 object categories.
2. VGG-19 is a deep-CNN architecture that is used to classify images. It has 19 layers; 16 are convolution layers,


**Fig. 1** Images from synthetic hazy datasets generated from the original Caltech-256 dataset [42] and COCO [14]

and three are fully connected. Furthermore, it has six MaxPool layers, one SoftMax layer, and 19.6 million FLOPS. The VGG-19 can classify images into 1000 object categories.

3. ResNet, referred to as Residual Networks, is an orthodox neural network that is used as a backbone for many computer vision applications. The proposed model won the ImageNet challenge and notably enables the training of very deep neural networks (150 + layers). Before ResNet, training deep neural networks was complicated due to problems such as vanishing gradients. However, ResNet has successfully handled this trade-off. In our empirical study, we uses ResNet-50, which can train 50 deep convolutional neural network layers.

To train the neural network, we used the pre-trained weights of VGG-16, VGG-19, and ResNet with the ImageNet database. Training involved using the bottleneck features of the pre-trained networks, initially loading the convolutional part of the model's pre-trained weights. The datasets (both training and testing) were subsequently fed to loaded weights to extract useful features. Then, the extracted features were saved in the workspace. The fully connected layers of the neural network (NN) architecture were then trained on the extracted features. Our training procedure efficiently used these pre-trained weights on a custom dataset, optimizing both training and testing. Table 1 and Fig. 2 demonstrate the performance of the trained classifiers on the clear and hazy test sets, showing that haze significantly affects the performance of the classification models. Our observations reveal that VGG-19 is more sensitive to hazy images than VGG-16, and ResNet-50 is the least accurate of the evaluated models.

Analyzing the effects of dehazing on classification models requires dehazing models. For this purpose, we employed the following dehazing models:

i. AOD-Net [41] is a lightweight and effective end-to-end dehazing neural network that reformulates the atmospheric scattering model. It has attracted considerable attention from the research community, likely because it does not need to separately estimate parameters such as atmospheric and transmission maps, and it produces haze-free images using a lightweight CNN.
ii. GCANet [44] is an end-to-end gated context aggregation network that removes gridding artifacts and directly restores haze-free images, with the proposed method applying a smoothed dilation technique and leveraging a gated sub-network to fuse features from different levels. The authors validated the method's capacity to surpass previous state-of-the-art methods quantitatively and qualitatively.
iii. FFANet [45] comprises three key components: a channel-attention-with-pixel-attention mechanism, expansion of the CNN's representational ability, and a feature fusion structure. The method surpasses previous state-of-the-art single-image dehazing methods by a high margin, both quantitatively and qualitatively.
iv. PFFNet [46] employs an encoder-decoder deep network via progressive feature fusions, directly learning a nonlinear transformation function by considering hazy input and ground truth images. The proposed method achieved superior dehazing results and could restore hazy images to up to 4 K resolution.

Tables 2, 3, 4 and 5 make apparent that applying dehazing directly to hazy images has diverse effects. For instance, in most cases, the deep-learning-based dehazing algorithms improved the accuracy of classification

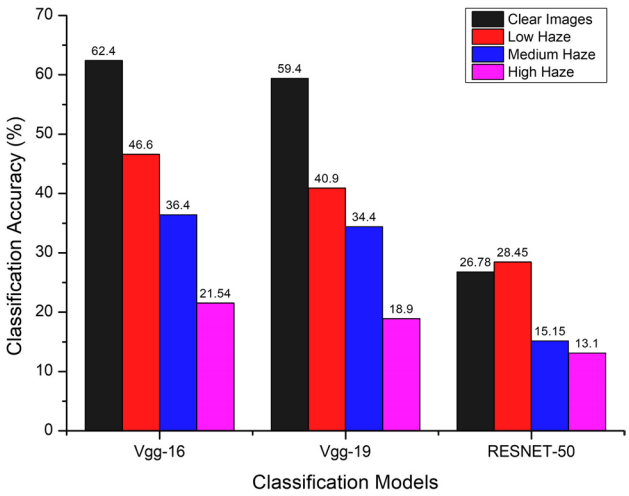



**Fig. 2** Performance of the three classification models on clear and synthetic hazy images

**Table 1** Accuracy of classification models on clear and hazy data

| Classification Models | Accuracy on Clear Images | On Low-Level Hazy Images | On Medium-Level Hazy Images | On High-Level Hazy Images |
|---|---|---|---|---|
| VGG-16 | 62.4 | 46.60 | 36.40 | 21.54 |
| VGG-19 | 59.4 | 40.90 | 34.40 | 18.90 |
| RESNET-50 | 26.78 | 28.45 | 15.15 | 13.10 |

**Table 2** Classification accuracy after removing haze using AOD-Net [41]

| Classification Models | Accuracy on Clear Images | On Low-Level Hazy Images | On Medium-Level Hazy Images | On High-Level Hazy Images |
|---|---|---|---|---|
| VGG-16 | 62.4 | 57.11 | 53.46 | 39.63 |
| VGG-19 | 59.4 | 51.63 | 48.78 | 33.74 |
| RESNET-50 | 26.78 | 24.18 | 20.32 | 16.26 |

**Table 3** Classification accuracy after removing haze using GCANet [44]

| Classification Models | Accuracy on Clear Images | On Low-Level Hazy Images | On Medium-Level Hazy Images | On High-Level Hazy Images |
|---|---|---|---|---|
| VGG -16 | 62.4 | 57.93 | 58.30 | 53.05 |
| VGG -19 | 59.4 | 53.86 | 52.25 | 52.04 |
| RESNET-50 | 26.78 | 23.58 | 23.68 | 21.95 |

**Table 4** Classification accuracy after removing haze using FFANet [45]

| Classification Models | Accuracy on Clear Images | On Low-Level Hazy Images | On Medium-Level Hazy Images | On High-Level Hazy Images |
|---|---|---|---|---|
| VGG -16 | 62.4 | 48.57 | 43.69 | 28.46 |
| VGG -19 | 59.4 | 43.50 | 38.62 | 24.8 |
| RESNET-50 | 26.78 | 14.02 | 09.9 | 05.80 |

**Table 5** Classification accuracy after removing haze using PFFNet [46]

| Classification Models | Accuracy on Clear Images | On Low-Level Hazy Images | On Medium-Level Hazy Images | On High-Level Hazy Images |
|---|---|---|---|---|
| VGG -16 | 62.4 | 56.91 | 55.08 | 48.78 |
| VGG -19 | 59.4 | 54.88 | 51.83 | 43.69 |
| RESNET-50 | 26.78 | 12.3 | 09.15 | 02.30 |

**Table 6** Classification accuracy after removing haze using BCCR [47]

| Classification Models | Accuracy on Clear Images | On Low-Level Hazy Images | On Medium-Level Hazy Images | On High-Level Hazy Images |
|---|---|---|---|---|
| VGG -16 | 62.4 | 42.61 | 39.05 | 28.98 |
| VGG -19 | 59.4 | 41.38 | 34.77 | 33.39 |
| RESNET-50 | 26.78 | 13.85 | 11.91 | 06.48 |

models when applied on hazy images. We have also included the following four conventional dehazing algorithms in our experiments: Boundary Constraint and Contextual Regularization (BCCR) [47], Fast Visibility Restoration (FVR) [48], Dark-Channel Prior (DCP) [49], and Color Attenuation Prior (CAP) [50]. Tables 6, 7, 8 and 9 present quantitative analysis of these dehazing algorithms.

Figure 3(a) indicates that dehazing improves the classification accuracy of VGG-16 and VGG-19, with GCANet performing better than the other dehazing algorithms. Nonetheless, AOD-Net, PFFNet, and FFANet also improved the accuracy of VGG-16 and VGG-19. Regarding ResNet-50's accuracy, we observed an irregularity in its mean average precision (mAP), its accuracy increased when it encountered low-level hazy images. Nonetheless, dehazing using AOD-Net and GCANet improved the accuracy of ResNet-50 for medium-level and high-level hazy images. However, in comparison, neither FFANet nor PFFNet produced any notable improvement.

**Table 7** Classification accuracy after removing haze using FVR [48]

| Classification Models | Accuracy on Clear Images | On Low-Level Hazy Images | On Medium-Level Hazy Images | On High-Level Hazy Images |
|---|---|---|---|---|
| VGG -16 | 62.4 | 33.8 | 36.32 | 26.98 |
| VGG -19 | 59.4 | 25.97 | 29.16 | 24.40 |
| RESNET-50 | 26.78 | 13.3 | 12.28 | 10.93 |

**Table 8** Classification accuracy after removing haze using DCP [49]

| Classification Models | Accuracy on Clear Images | On Low-Level Hazy Images | On Medium-Level Hazy Images | On High-Level Hazy Images |
|---|---|---|---|---|
| VGG -16 | 62.4 | 31.92 | 26.92 | 17.57 |
| VGG -19 | 59.4 | 35.51 | 37.11 | 19.38 |
| RESNET-50 | 26.78 | 05.91 | 14.35 | 08.06 |

**Table 9** Classification accuracy after removing haze using CAP [50]

| Classification Models | Accuracy on Clear Images | On Low-Level Hazy Images | On Medium-Level Hazy Images | On High-Level Hazy Images |
|---|---|---|---|---|
| VGG -16 | 62.4 | 37.89 | 41.29 | 22.83 |
| VGG -19 | 59.4 | 36.49 | 40.91 | 29.64 |
| RESNET-50 | 26.78 | 09.72 | 10.96 | 11.14 |

In contrast, Fig. 3(b) indicates that conventional dehazing algorithms mostly did not substantially improve the trained model's classification accuracy, supporting the findings of Pei et al. [8]. Furthermore, no dehazing algorithm enabled the classification models to match their original (clear image) classification accuracy when applied directly as a preprocessing step.

## 4 Haze and dehazing analysis for detection tasks

For detection tasks, we performed the haze and dehazing analysis using two procedures. First, we used pre-trained detection weights on clear images (COCO) [14] before providing the pre-trained models with synthesized hazy and dehazed images and then analyze their effects. However, given the complications associated with synthesizing a large-scale hazy dataset and then adding corresponding annotations, we synthesized a total of 300 images from the COCO validation set (2017) at three levels of haze (i.e., low-, medium-, and high-level). For synthesized hazy images, we considered the annotations of the corresponding clear images (i.e., the original images). For dehazed images, we also considered the annotations associated with the original images.

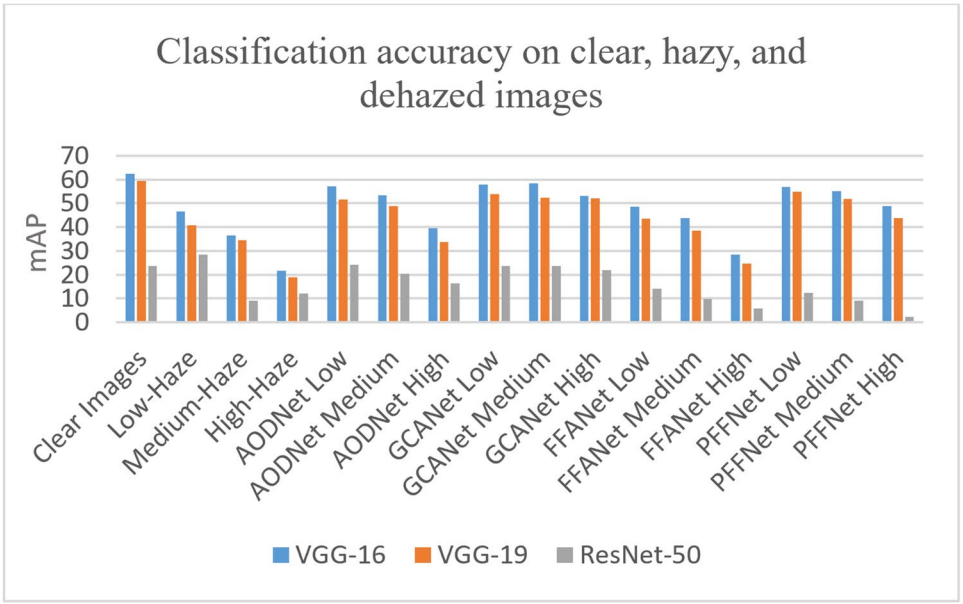
Second, the pre-trained models were trained on a hazy real-world task-driven testing set (RTTS) adopted from RESIDE [51] and containing real-world hazy images with available annotations. One hundred images were dehazed from their test set and provided to the trained models for further evaluation. The subsequent sections detail the experiments and their results.

### 4.1 Case 1

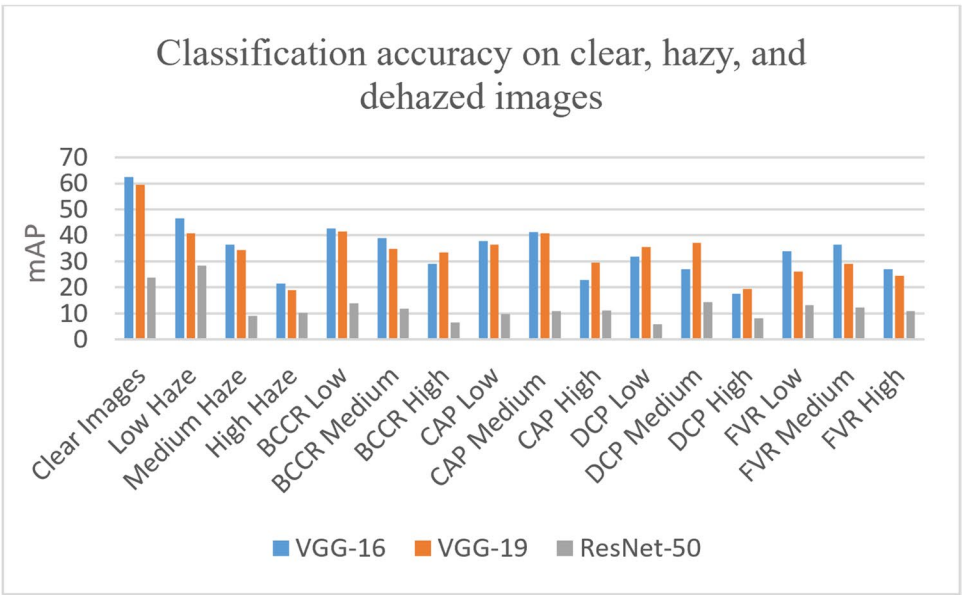
#### 4.1.1 Experimental procedure and results

To determine haze and dehazing effects on detection models, we selected three state-of-the-art object detection algorithms: FRCNN [27], YOLOv3 [31], and EfficientDet [32].

1. FRCNN [27] uses an RPN, which shares full-image convolutional features with the detection network and simultaneously performs the object bounds prediction and objectness score for every position. The proposed technique merges the RPN with Fast R-CNN [26] to create a single network with shared convolutional features. The proposed model has a frame rate of 5 fps on a GPU and has demonstrated state-of-the-art accuracy on large-scale datasets with 300 proposals per image. Our implementation used the built-in implementation of FRCNN




(a)




(b)

**Fig. 3** Accuracy of classification models with respect to haze and dehazing

with the torch-vision library in PyTorch. The FRCNN achieved 49.91% mAP on clear COCO images.

2. YOLOv3 [31] is part of a popular line of real-time object detection systems. The first version was introduced by Joseph Redmon [29]. On a Pascal Titan X, it processes images at 30 fps and demonstrates 57.9% mAP on the COCO test-dev. This first version was then followed by YOLO 9000 [30], YOLOv3 [31], and YOLOv4 [52]. YOLOv3 is the 3rd generation of the YOLO family, and Pjreddie trained it on multiple high-end GPUs, including the Tesla V100. Using the pre-trained weights on the COCO dataset in this study, the model achieved 63.94% mAP on clear images.
3. EfficientDet [32] is a recent state-of-the-art detection model based on advanced optimizations and backbones and includes a weighted bi-directional feature pyramid network, which multi-scales the feature fusion robustly, and a compound scaling technique, which simultaneously scales the resolution, depth, and width for all backbone, feature networks, and box/class prediction networks. Due to advanced optimizations and better backbones, it achieved better efficiency and

demonstrated 55.1% mAP on the COCO test-dev. It is 4x–9 × smaller and uses 13x–42 × fewer FLOPs than previous detectors, as well as running 2x–4 × faster on GPUs and 5x–11 × faster on CPUs than other detectors. EfficientDet is trained by Google using multiple high-end GPUs, including the Tesla V100. Using Google's pre-trained weights on COCO, the system achieved 75.06% mAP on clear images.

Figure 4(a) compares the mAP scores on clear images of FRCNN, YOLOv3, and EfficientDet. The FRCNN demonstrated 49.91% mAP and YOLOv3 63.94%; however, the EfficientDet performed best, achieving 75.06% mAP.

Figure 4(b) shows the mAP scores achieved by the three systems when presented with synthesized hazy images (from the three haze levels) in comparison with the scores achieved for clear images. The FRCNN's mAP reduced from 49.9% to 43.72% in the presence of low-level haze. For medium-level and high-level haze, its accuracy dropped to 37.54% and 30.91%, respectively. Meanwhile, for YOLOv3, the mAP dropped from 63.94% to 57.95% when considering low-level haze and reduced to 55.87% and 50.39%, respectively, when it encountered medium- and high-level haze. Haze also impacted EfficientDet's detection accuracy, although, it was less affected than the other two detection models.

Figure 5 shows the visual degradation analysis of the detection models in the presence of haze. Next, we dehazed the synthesized hazy images using AOD-Net, GCANet, FFANet, and PFFNet to analyze the impact of dehazing on the detection models. These dehazing algorithms were chosen again because they have demonstrated better performance than BCCR [47], FVR [48], DCP [49], and CAP [50] in the classification analysis Sect. 3.

Figure 6(a) reveals that AOD-Net improved the detection performance of FRCNN and YOLOv3 for low-, medium-, and high-level hazy images. Meanwhile, upon providing the AOD-Net dehazed test data to EfficientDet, we noted slight and variable changes in the mAP scores. For instance, the mAP score decreased for low-level hazy images, for medium-level hazy images, although it was almost the same mAP as for clear images, and it slightly improved in the case of high-level hazy images.

Figure 6(b) shows the mAP scores of the three detection models following GCANet dehazing. Dehazing with GCANet improved FRCNN's mAP scores for all three levels of haziness. For YOLOv3, the mAP scores slightly progressed for low-, medium-, and high-level hazy images. However, GCANet dehazing did not improve the performance of EfficientDet, with the mAP scores slightly decreasing.

Figure 6(c) shows that dehazing using FFANet worsened the performances of FRCNN and EfficientDet for all hazy levels. Similarly, it did not improve the performance of YOLOv3 except for medium-level hazy images.

Finally, Fig. 6(d) shows that dehazing using PFFNet improved the mAP scores of FRCNN. For YOLOv3, the mAP increased slightly for medium- and high-level hazy images but decreased for low-level hazy images. Meanwhile, for EfficientDet, lower mAP scores were recorded for all hazy levels.

Figure 7 generalizes the experimental results for the detection and dehazing models, demonstrating that haze affected the detection tasks, with FRCNN most impacted and YOLOv3 the second-most impacted, with EfficientDet least affected. Neither FFANet nor PFFNet dehazing improved the mAP scores for all detection models, with PFFNet dehazing surpassing only low-level hazy image accuracy and only by a small margin. Similarly, for YOLOv3, FFANet dehazing only improved medium-level hazy image accuracy only by 2%.

Meanwhile, AOD-Net and GCANet dehazing both improved accuracy at all three levels of haziness for both FRCNN and YOLOv3. However, for EfficientDet, none of the accuracy improved for any hazy level. We noted that applying the dehazing algorithms directly with detection models did not improve or match the clear images mAP scores except for the case of AOD-Net dehazing of low-level hazy images' for FRCNN detector (Figs. 8 and 9).

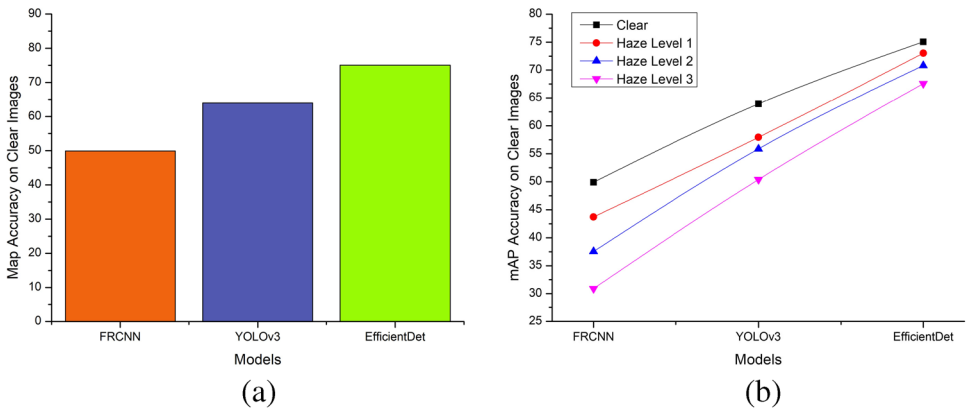



**Fig. 4** (a) Comparison of mAP scores on clear images. (b) Comparison of mAP scores in the presence of haze

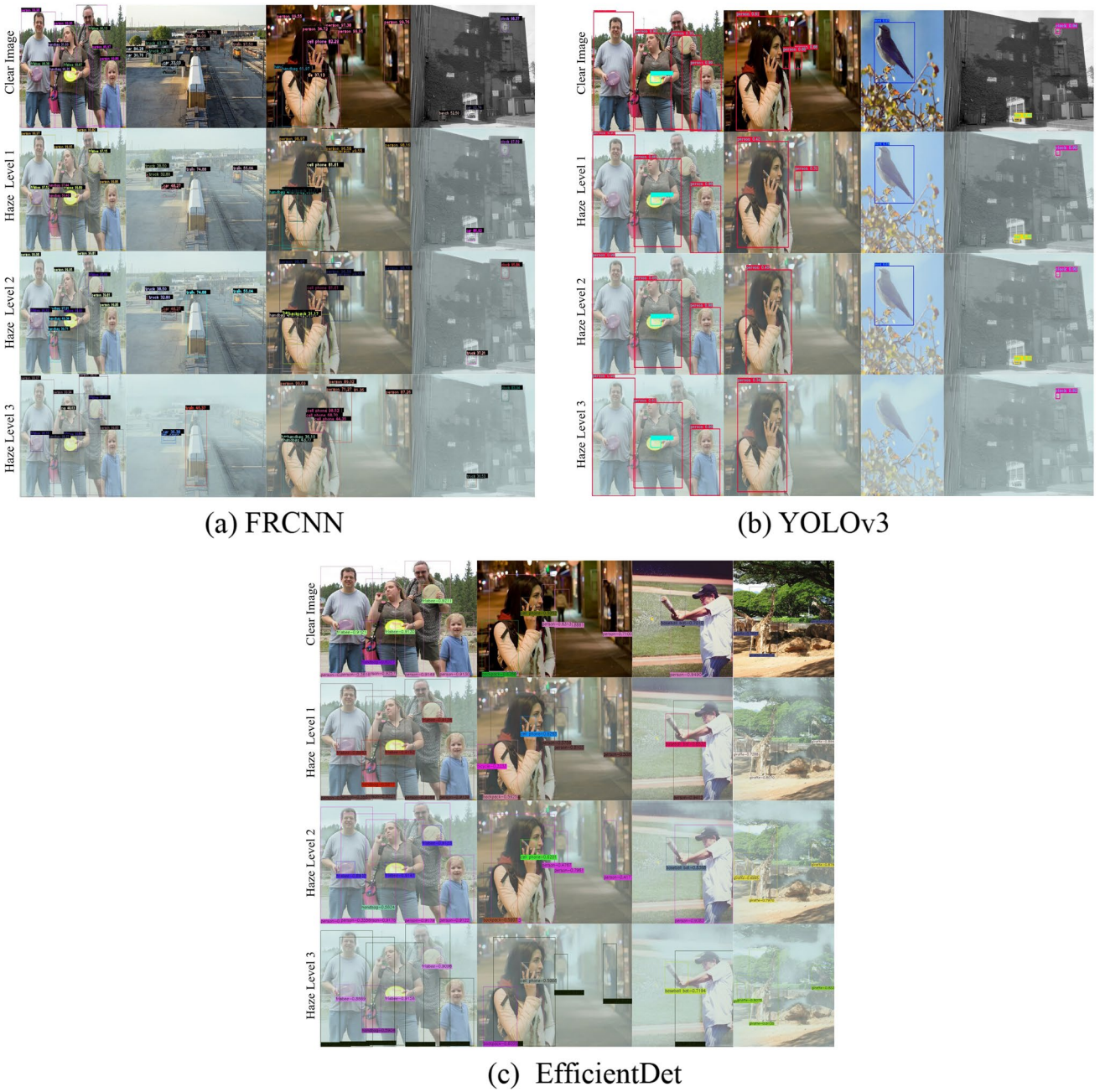



**Fig. 5** Visual analysis of detection models demonstrating performance degradation in the presence of haze

### 4.2 Case 2

The second application involved training and testing the same detection models as in Case 1 (i.e., FRCNN, YOLOv3, and EfficientDet) on RTTS [51], and again dehazing the RTTS dataset using AOD-Net, GCANet, PFFNet, and FFANet. The trained models were further evaluated on the dehazed images to identify the impact of dehazing.

The RTTS dataset has only five classes (car, bus, person, motorbike, and bicycle) and the annotations are provided in the standard VOC dataset format (i.e., XML files for each image with classes and bounding boxes). Given the dataset only includes about 4,322 images, thus, we adopted the transfer learning training strategy.

We trained the D4 variant of EfficientDet and used the pre-trained weights on the ImageNet dataset. The training was conducted for ten epochs with an SGD optimizer. Next,

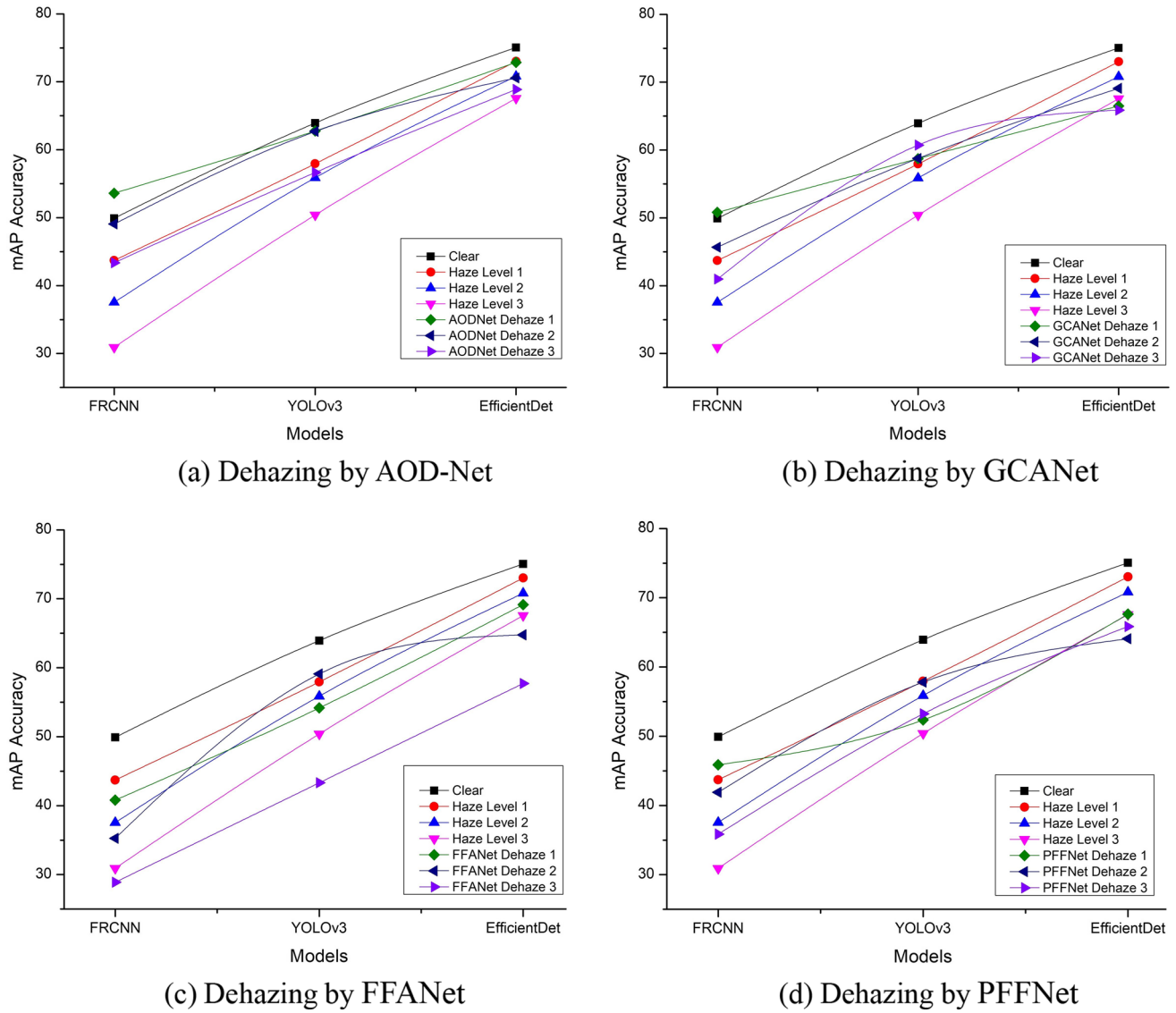



**Fig. 6** mAP scores of detection models for dehazed images

FRCNN was trained with a ResNet-50 backbone using the pre-trained weights on the COCO dataset. The training was conducted for 15 epochs with an SGD optimizer. Finally, YOLOv3 was trained with a Darknet backbone using the pre-trained weights on the COCO dataset. The training was conducted for 15 epochs with an ADAM optimizer. An Nvidia P100 GPU on Google Colab was used for training and testing.

Figure 10 provides the quantitative analysis of haze and dehazing and indicates that, compared to YOLOv3, EfficientDet and FRCNN performed well on hazy images. It also makes apparent that GCANet and FFANet performed well by achieving mAP scores close to hazy-trained scores. However, none of the dehazing algorithms helped the detection models to improve the hazy-trained accuracy, as Fig. 10 illustrates. Figures 11, 12 and 13 present the visual detection results.

## 5 Discussion

This study confirms that deep neural architectures are vulnerable to haze, and it is challenging to perform classification tasks in the presence of haze. Adopting the transfer learning strategy with bottleneck features and then applying dehazing algorithms—namely, AOD-Net, GCANet, PFFNet, and FFANet—improved the mAP scores for the VGG-16 and VGG-19 classification models. However, conventional dehazing algorithms—namely, BCCR, CAP, DCP, and FVR—did not substantially improve the performance of classification

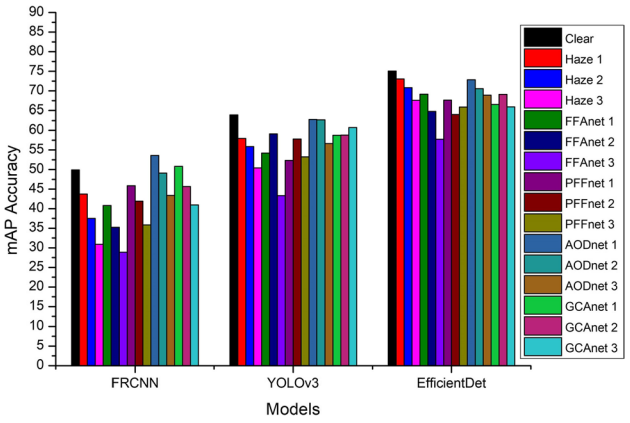



**Fig. 7** Accuracy comparison of detection models on clear, hazy, and dehazed images

models, supporting the findings of Pei et al. [8]. The VGG-16 and VGG-19 likely performed well because of their deep layers. Notably, most of the synthetic hazy Caltech-256 images have frontal views of objects and for that the deep learning-based dehazing methods (AOD-Net, GCANet, PFFNet, and FFANet) have demonstrated better dehazing results. The ResNet-50 classification model did not achieve substantial accuracy when applied to clear, hazy, or dehazed images.

Our detection Case 1 experiments demonstrated that detection models are also vulnerable to haze and have greater difficulty completing detection tasks when encountering noisy images. For example, in the presence of haze, the accuracy of all detection models decreased compared to their accuracy in the context of clear images, with FRCNN most impacted, followed by YOLOv3, with EfficientDet least affected. In terms of dehazing, although we noted that AOD-Net and GCANet improved the detection performance of both FRCNN and YOlOv3, using both dehazing algorithms with EfficientDet produced only minor changes in mAP values. Next, PFFNet improved the accuracy of FRCNN for all hazy levels, while it slightly improved the mAP scores of YOLOv3 for medium- and high-level hazy images and decreased the mAP score for low-level hazy images. Finally, FFANet's dehazing performance was noted weak compared to the other three dehazing algorithms.

In detection Case 2, in which we trained the detection models on RTTS dataset, the FRCNN and EfficientDet performed better than YOLOv3. It was expected that the accuracy of the detection models would increase, although none of the four dehazing algorithms had a positive impact. It should be noted that the RTTS dataset contained noisy images with varying levels of haze, that is, where some images featured very low haze, while other images were very hazy. This led the detection models to achieve accuracies near to their hazy-trained mAP but preventing them surpassing those performances.

Furthermore, the RTTS was a highly imbalanced dataset, featuring more than 20,000 samples of cars and about 10,000 samples of people, with the other classes featuring between 2,000 and 3,000 images. We believe that there should be more annotated classes and instances because, using the RTTS dataset, the detection models could detect objects from other classes as, for example, cars (i.e., false positives), as Fig. 14 shows.

Table 10 provides a broad overview of the experimental results, answering the question of whether dehazing improves the performance of vision models when applied directly as a preprocessing step. The overview indicates that AODNet, GCANet, PFFNet, and FFANet dehazing improved the accuracy of the classification models, with AOD-Net and GCANet more effective than FFANet and PFFNet. Meanwhile, BCCR, CAP, DCP, and FVR did not substantially improve the performances of the classification models.

In terms of detection Case 1, the dehazing algorithms––namely, AOD-Net, GCANet, and PFFNet––improved the mAP scores for FRCNN and YOLOv3. However, these dehazing algorithms performed poorly when applied with EfficientDet, and can lead to lower mAP scores. Meanwhile, in detection Case 2, the dehazing algorithms failed to improve the performance of any of the detection models.

There are several facts that high-level vision models did not perform very well though testing them on the dehazed images. First, the detection and classification tasks might have been too complex. Second, dehazing did not completely restore a hazy image to their haze-free version. Furthermore, both the haze addition and dehazing processes changes much of the image information considerably, introducing color shifts and distortions. This might have led the models to identify many objects in the dehazed test images as outliers or make otherwise false predictions. Likewise, using dehazing as a preprocessing step does not introduce such new information, which could improve the performance of high-level vision models [8].

Simply using more classy detection models will not improve the dehazing-detection cascade performance due to the domain gap between hazy/dehazed and clean images (on which typical detectors are trained) [53]. Thus, it is necessary to overcome the domain gap to improve efficient and reliable detection. To address the haze problem for vision systems, it is also essential to try various combinations of influential dehazing/detection models, such as using a cascade of the AOD-Net dehazing model with the FRCNN detection model to detect objects in hazy images. However, such a cascade must be subjected to joint optimization [41, 54, 55]. Adopting the cascading strategy will improve dehazing performance and enhance the accuracy of high-level vision models. Meanwhile, constructing new


**Fig. 8** Visual analysis of the performances of FRCNN and YOLOv3 with dehazing algorithms


**Fig. 9** Visual analysis of the performances of YOLOv3 and EfficientDet with dehazing algorithms

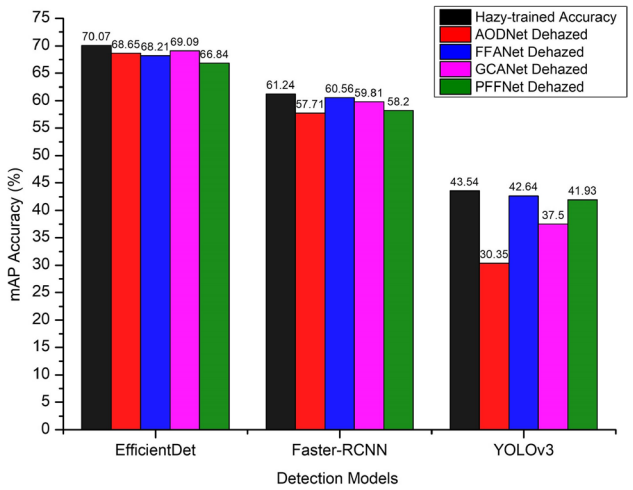



**Fig. 10** mAP scores of the detection algorithms for hazy and dehazed images

loss functions represents another approach to strengthen the accuracy of vision models in the case of noisy images [56].

Our haze and dehazing analysis demonstrated the need for particular consideration of image degradation effects, which significantly impact high-level vision tasks. There is also a shortage of proper (i.e., featuring more classes and annotations) and large-scale outdoor hazy datasets for detection tasks. Currently, RTTS is the only authentic hazy dataset of the RESIDE [51] benchmark that could be utilized for training and testing vision models. However, in its current form, RTTS represents an imbalanced dataset that is relatively unsuitable for vision model evaluation, often resulting in false positives, and requires more diversity in terms of images, classes, and annotations.

## 6 Conclusions

Most deep-learning-based vision models are trained and validated using clear images, avoiding noisy, or hazy, images. However, such models are likely to encounter degraded images affected by haze, mist, and fog in real-time applications. Therefore, we have performed extensive experiments to study the effects of haze and dehazing on both detection and classification tasks. Few classification and detection models are trained and validated using hazy images, haze-free images, synthesized hazy images, and dehazed images. Our findings indicate that haze considerably affects high-level vision tasks. For classification tasks, few dehazing algorithms improved classification model's accuracy to a trivial extent. Furthermore, in some instances, dehazing degraded or did not enhance classification accuracy. For detection tasks, we performed our experiments in two

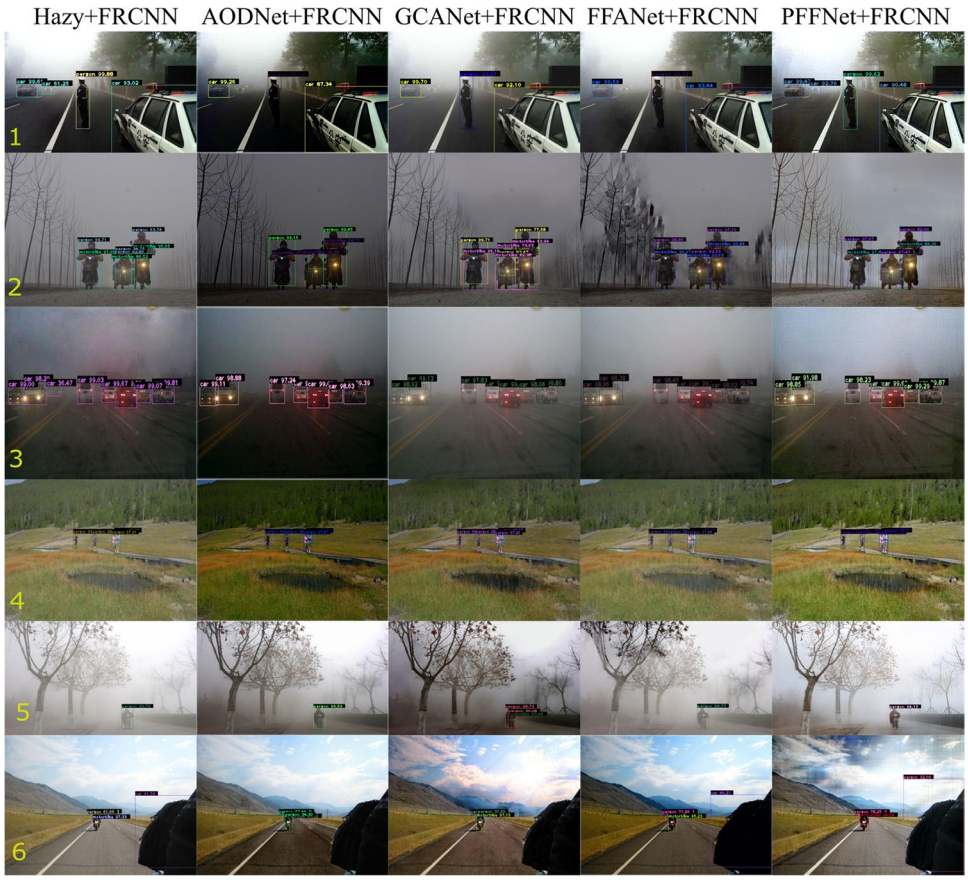



**Fig. 11** Visual analysis of the performance of FRCNN with dehazed images


**Fig. 12** Visual analysis of the performance of YOLOv3 with dehazed images


**Fig. 13** Visual analysis of the performance of EfficientDet with dehazed images

settings. Initially, we used the pre-trained weights on clear images and then tested hazy and dehazed images on the pre-trained models to analyze their impact. Using dehazing as a preprocessing step with detection models partially improved their accuracy. Second, we trained the three detection models using a hazy benchmark dataset and then tested them on dehazed images. In this case, the dehazing algorithms failed to improve the accuracy of the trained models.

Our experimental findings confirm that deep neural architectures are vulnerable to haze, and that it is hard for them to


**Fig. 14** False positives detections

**Table 10** Overview of experimental outcomes

| Classification Task | | | | | |
|---|---|---|---|---|---|
| Classification Model | Hazy Levels | BCCR | CAP | DCP | FVR |
| | Low-level haze | mAP deceased | mAP decreased | mAP decreased | mAP decreased |
| VGG-16 | Medium-level haze | mAP increased | mAP increased | mAP decreased | almost same mAP |
| | High-level haze | mAP increased | mAP increased | mAP decreased | mAP increased |
| | Low-level haze | mAP increased | mAP decreased | mAP decreased | mAP decreased |
| VGG-19 | Medium-level haze | almost same mAP | mAP increased | mAP increased | mAP decreased |
| | High-level haze | mAP increased | mAP increased | mAP decreased | mAP increased |
| | Low-level haze | mAP decreased | decreased | mAP decreased | mAP decreased |
| ResNet-50 | Medium-level haze | mAP increased | almost same mAP | mAP increased | mAP increased |
| | High-level haze | mAP decreased | almost same mAP | mAP decreased | almost same mAP |
| | | AOD-Net | GCANet | FFANet | PFFNet |
| | Low-level haze | mAP increased | mAP increased | mAP increased | mAP increased |
| VGG-16 | Medium-level haze | mAP increased | mAP increased | mAP increased | mAP increased |
| | High-level haze | mAP increased | mAP increased | mAP increased | mAP increased |
| | Low-level haze | mAP increased | mAP increased | mAP increased | mAP increased |
| VGG-19 | Medium-level haze | mAP increased | mAP increased | mAP increased | mAP increased |
| | High-level haze | mAP increased | mAP increased | mAP increased | mAP increased |
| | Low-level haze | mAP decreased | mAP decreased | mAP decreased | mAP decreased |
| ResNet-50 | Medium-level haze | mAP increased | mAP increased | mAP decreased | mAP decreased |
| | High-level haze | mAP increased | mAP increased | mAP decreased | mAP decreased |
| Detection Model | | Detection Task (Case-1) | | | |
| | Low-level haze | mAP increased | mAP increased | mAP decreased | mAP increased |
| FRCNN | Medium-level haze | mAP increased | mAP increased | mAP decreased | mAP increased |
| | High-level haze | mAP increased | mAP increased | mAP decreased | mAP increased |
| | Low-level haze | mAP increased | mAP slightly increased | mAP decreased | mAP decreased |
| YOLOv3 | Medium-level haze | mAP increased | mAP slightly increased | mAP increased | mAP slightly increased |
| | High-level haze | mAP increased | mAP increased | mAP decreased | mAP slightly increased |
| | Low-level haze | mAP decreased | mAP decreased | mAP decreased | mAP decreased |
| EfficientDet | Medium-level haze | almost same mAP | mAP decreased | mAP decreased | mAP decreased |
| | High-level haze | mAP slightly increased | mAP decreased | mAP decreased | mAP decreased |
| Detection Models | | For Detection Task (Case-2) | | | |
| FRCNN | Low-level haze | mAP decreased | mAP decreased | mAP decreased | mAP decreased |
| YOLOv3 | Medium-level haze | mAP decreased | mAP decreased | mAP decreased | mAP decreased |
| EfficientDet | High-level haze | mAP decreased | mAP decreased | mAP decreased | mAP decreased |

accurately classify and detect in the presence of haze. Notably, using dehazing directly as a preprocessing step for high-level vision tasks represents an unpredictable procedure that does not lead to substantial improvements. However, dehazing has limited potential to improve performance on classification tasks, even if robust dehazing algorithms are used. To improve the performance of vision models in hazy weather conditions, a dehazing-detection cascaded framework (subjected to joint optimization) is a better approach. Adopting such a strategy would improve the dehazing network's performance and, ultimately, model accuracy. Therefore, future research should focus on addressing such challenges, especially in the context of developing new high-level vision models.

**Acknowledgements** This work was supported by National Science Foundation of China (NSFC-61860206004).

## Declarations

**Competing of interest** The authors whose names are listed declare no conflict of interest.

## References

1. Wu H, Qu Y, Lin S, Zhou J, Qiao R, Zhang Z, ... Ma L (2021) Contrastive learning for compact single image dehazing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 10551–10560)
2. Zhang S, He F, Ren W, Yao J (2020) Joint learning of image detail and transmission map for single image dehazing. Vis Comput 36(2):305–316
3. Zhang J, He F, Chen Y (2020) A new haze removal approach for sky/river alike scenes based on external and internal clues. Multimedia Tools and Applications 79(3):2085–2107
4. Yoon KJ, Shyam P, Kim KS (2021, February) Towards domain invariant single image dehazing. In AAAI Conference on Aritificial Intelligence. Association for the Advancement of Artificial Intelligence
5. Wang C, Huang Y, Zou Y, Xu Y (2021, June) FWB-Net: front white balance network for color shift correction in single image dehazing via atmospheric light estimation. In ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 2040–2044). IEEE.
6. Yi X, Ma B, Zhang Y, Liu L, Wu J (2021) Two-step image dehazing with intra-domain and inter-domain adaptation. arXiv preprint arXiv:2102.03501
7. Pei Y, Huang Y, Zou Q, Zang H, Zhang X, Wang S (2018) Effects of image degradations to cnn-based image classification. arXiv preprint arXiv:1810.05552
8. Pei, Y., Huang, Y., Zou, Q., Lu, Y., & Wang, S. (2018). Does haze removal help cnn-based image classification?. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 682–697).
9. Zhang S, He F, Ren W (2020) NLDN: Non-local dehazing network for dense haze removal. Neurocomputing 410:363–373
10. Zhang S, He F (2020) DRCDN: learning deep residual convolutional dehazing networks. Vis Comput 36(9):1797–1808
11. Yang, J., Yu, K., Gong, Y., & Huang, T. (2009, June). Linear spatial pyramid matching using sparse coding for image classification. In 2009 IEEE Conference on computer vision and pattern recognition (pp. 1794–1801). IEEE.
12. Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248–255). Ieee.
13. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A (2010) The pascal visual object classes (voc) challenge. Int J Comput Vision 88(2):303–338
14. Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014, September). Microsoft coco: Common objects in context. In European conference on computer vision (pp. 740–755). Springer, Cham.
15. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 25:1097–1105
16. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
17. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1–9).
18. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770–778).
19. Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.
20. Liu, L., Shen, C., & Van Den Hengel, A. (2015). The treasure beneath convolutional layers: Cross-convolutional-layer pooling for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4749–4757).
21. Wang, J., Yang, Y., Mao, J., Huang, Z., Huang, C., & Xu, W. (2016). Cnn-rnn: A unified framework for multi-label image classification. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2285–2294).
22. Durand, T., Mordan, T., Thome, N., & Cord, M. (2017). Wildcat: Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 642–651).
23. Wang, F., Jiang, M., Qian, C., Yang, S., Li, C., Zhang, H., ... & Tang, X. (2017). Residual attention network for image classification. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3156–3164).
24. Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 580–587).
25. He K, Zhang X, Ren S, Sun J (2015) Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell 37(9):1904–1916
26. Girshick, R. (2015). Fast r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 1440–1448).
27. Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: Towards real-time object detection with region proposal networks. Adv Neural Inf Process Syst 28:91–99
28. Yu, J., Jiang, Y., Wang, Z., Cao, Z., & Huang, T. (2016, October). Unitbox: An advanced object detection network. In Proceedings of the 24th ACM international conference on Multimedia (pp. 516–520).

29. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779–788).
30. Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7263–7271).
31. Redmon, J., & Farhadi, A. (1804). YOLOv3: an incremental improvement (2018). arXiv preprint arXiv:1804.02767, 20.
32. Tan, M., Pang, R., & Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 10781–10790).
33. Ren CX, Dai DQ, Yan H (2012) Coupled kernel embedding for low-resolution face image recognition. IEEE Trans Image Process 21(8):3770–3783
34. Zou WW, Yuen PC (2011) Very low resolution face recognition problem. IEEE Trans Image Process 21(1):327–340
35. Basu S, Karki M, Ganguly S, DiBiano R, Mukhopadhyay S, Gayaka S, ... Nemani R (2017) Learning sparse feature representations using probabilistic quadtrees and deep belief nets. Neural Processing Letters, 45(3), 855-867
36. Xiao H, Rasul K, Vollgraf R (2017) Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747
37. Karam LJ, Zhu T (2015, March) Quality labeled faces in the wild (QLFW): a database for studying face recognition in real-world environments. In Human Vision and Electronic Imaging XX (Vol. 9394, p. 93940B). International Society for Optics and Photonics
38. Roy P, Ghosh S, Bhattacharya S, Pal U (2018) Effects of degradations on deep neural network architectures. arXiv preprint arXiv:1807.10108
39. Wang Z, Chang S, Yang Y, Liu D, Huang TS (2016) Studying very low resolution recognition using deep networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4792–4800)
40. Diamond S, Sitzmann V, Boyd S, Wetzstein G, Heide F (2017) Dirty pixels: Optimizing image classification architectures for raw sensor data. arXiv e-prints, arXiv-1701
41. Li, B., Peng, X., Wang, Z., Xu, J., & Feng, D. (2017). Aod-net: All-in-one dehazing network. In Proceedings of the IEEE international conference on computer vision (pp. 4770–4778).
42. Griffin, G., Holub, A., & Perona, P. (2007). Caltech-256 object category dataset.
43. Zhang, N., Zhang, L., & Cheng, Z. (2017, November). Towards simulating foggy and hazy images and evaluating their authenticity. In International Conference on Neural Information Processing (pp. 405–415). Springer, Cham.
44. Chen, D., He, M., Fan, Q., Liao, J., Zhang, L., Hou, D., ... & Hua, G. (2019, January). Gated context aggregation network for image dehazing and deraining. In 2019 IEEE winter conference on applications of computer vision (WACV) (pp. 1375–1383). IEEE.
45. Qin, X., Wang, Z., Bai, Y., Xie, X., & Jia, H. (2020, April). FFA-Net: Feature fusion attention network for single image dehazing. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, No. 07, pp. 11908–11915).
46. Mei, K., Jiang, A., Li, J., & Wang, M. (2018, December). Progressive feature fusion network for realistic image dehazing. In Asian conference on computer vision (pp. 203–215). Springer, Cham.
47. Meng, G., Wang, Y., Duan, J., Xiang, S., & Pan, C. (2013). Efficient image dehazing with boundary constraint and contextual regularization. In Proceedings of the IEEE international conference on computer vision (pp. 617–624).
48. Tarel, J. P., & Hautiere, N. (2009, September). Fast visibility restoration from a single color or gray level image. In 2009 IEEE 12th international conference on computer vision (pp. 2201–2208). IEEE.
49. He K, Sun J, Tang X (2010) Single image haze removal using dark channel prior. IEEE Trans Pattern Anal Mach Intell 33(12):2341–2353
50. Zhu Q, Mai J, Shao L (2015) A fast single image haze removal algorithm using color attenuation prior. IEEE Trans Image Process 24(11):3522–3533
51. Li B, Ren W, Fu D, Tao D, Feng D, Zeng W, Wang Z (2018) Benchmarking single-image dehazing and beyond. IEEE Trans Image Process 28(1):492–505
52. Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.
53. Liu, Y., Zhao, G., Gong, B., Li, Y., Raj, R., Goel, N., ... & Tao, D. (2018). Improved techniques for learning to dehaze and beyond: A collective study. arXiv preprint arXiv:1807.00202.
54. Liu, D., Wen, B., Liu, X., Wang, Z., & Huang, T. S. (2017). When image denoising meets high-level vision tasks: A deep learning approach. arXiv preprint arXiv:1706.04284.
55. Cheng, B., Wang, Z., Zhang, Z., Li, Z., Liu, D., Yang, J., ... & Huang, T. S. (2017, October). Robust emotion recognition from low quality and low bit rate video: A deep learning approach. In 2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII) (pp. 65–70). IEEE.
56. Carlini, N., & Wagner, D. (2017, November). Adversarial examples are not easily detected: Bypassing ten detection methods. In Proceedings of the 10th ACM workshop on artificial intelligence and security (pp. 3–14).

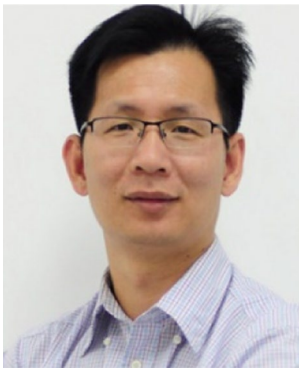



**Haseeb Hassan** received his Master's degree in Computer Science from International Islamic University, Islamabad (IIUI), Pakistan. He has been awarded a Ph.D. degree in Computer Science from Anhui University, Hefei, China. He is currently working as a Post-Doctoral candidate having affiliations with the College of Applied Sciences, Shenzhen University, Shenzhen, China, and College of Big Data and Internet, Shenzhen Technology University, Shenzhen, China. His current research interests are Computer Vision, Machine Learning, Deep Learning, Medical imaging, and he published several research articles related to his topics.

**Pranshu Mishra** received his B.S. degree in Computer Science from SJB Institute of Technology, India, in 2020. He is currently working as a Software Engineer at Amazon India. His main research interests include Image Processing, Computer vision, Machine Learning, and Deep Learning.

**Muhammad Ahmad** received the M.S. degree in electronics engineering from International Islamic University, Islamabad, Pakistan, in 2011, and the Ph.D. degree in computer science and engineering from Innopolis University, Innopolis, Russia, in 2019, and the second Ph.D. degree in cyber-physical systems from the University of Messina, Messina, Italy, in 2021.He is currently associated with the National University of Computer and Emerging Sciences (FAST-NUCES), Islamabad, Pakistan. He was an Assistant Professor, Lecturer, Instructor, Research Fellow, Research Associate, and Research Assistant with a number of international/national universities. He was with Ericsson (Mobilink Project) as Radio Access Network (RAN) Supervisor. He has authored and co-authored more than 70 scientific contributions to international journals, conferences, and books. He has supervised/co-supervised several graduates (M.S. and Ph.D.). His research interests include hyperspectral imaging, remote sensing, machine learning, computer vision, and wearable computing. Dr. Ahmad served/serving as a Lead/Guest Editor for several special issues in journals SCI/E and JCR. He has delivered a number of invited and keynote talks and reviewed (reviewing) the technology-leading articles for journals.

**Ali Kashif Bashir** received his Ph.D. in Computer Science and Engineering from Korea University, South Korea. Currently, he is a Reader of Networks and Security at the Department of Computing and Mathematics, Manchester Metropolitan University, United Kingdom. He also enjoys adjunct and honorary affiliations of the University of Electronics Science and Technology of China (UESTC), National University of Science and Technology, Islamabad (NUST), and the University of Guelph, Canada. He has authored over 200 research articles; and served as Co-PI from research bodies of South Korea, Japan, E.U., U.K., and the Middle East. His research interests include the internet of things, wireless networks, distributed systems, network/cyber security, network function virtualization, machine learning, etc. Currently, he is also serving as the Editor-in-chief of the IEEE FUTURE DIRECTIONS NEWSLETTER. He leads many conferences as a chair (program, publicity, and track) and has organized workshops in flagship conferences like IEEE Infocom, IEEE Globecom, IEEE Mobicom, etc. He has given more than 35 invited and keynote talks at international conferences across the globe.



**Bingding Huang** received a B.S degree in cell biology from the University of Science and Technology of China (USTC) in 2002. He received his M.S degree in Computer Science from Saarland University and Ph.D. in Computer Science from Dresden Technical University Germany in 2004 and 2008, respectively. Currently, he is a distinguished professor in Big Data and Bioinformatics at Shenzhen Technology University, China. He published several high-impact research articles in top-notch journals. His current research interests include Computer Vision, Machine Learning, Deep Learning, Medical imaging, Computational Biology, and Bioinformatics.

**Bin Luo** received his BEng. and MEng. degrees in electronics from Anhui University, China. In 2002, he was awarded a Ph.D. degree in Computer Science from the University of York, UK. He is currently a full professor at Anhui University. He is the Chair of the IEEE Hefei Subsection and an Associate Chair of IAPR TC15. He serves as the Editor-in-Chief (EiC) of the Journal of Anhui University (Natural Science Edition), an Associate Editor of several international journals, including Pattern Recognition, Pattern Recognition Letters, Cognitive Computation, and International Journal of Automation and Computing. He was the Guest Editor for the journal Special Issue (S.I.) of Pattern Recognition Letters (PRL) and Cognitive Computation. His current research interests include pattern recognition and digital image processing. In particular, he is interested in structural pattern recognition, graph-spectral analysis, image and graph matching. He has published more than 500 research papers in well-known journals, edited several books, and has been served as conferences Chairs. His papers are published in the IEEE TPAMI, IEEE TIP, Pattern Recognition, Pattern Recognition Letters (PRL), Neurocomputing, CVPR, NIPS, IJCAI, and AAAI with extensive citations.