Detection-driven exposure-correction network for nighttime drone-view object detection.

XI, Y., JIA, W., MIAO, Q., FENG, J., REN, J. and LUO, H.

2024

Detection-Driven Exposure-Correction Network for Nighttime Drone-View Object Detection

Yue Xi , Wenjing Jia , *Member, IEEE*, Qiguang Miao , *Senior Member, IEEE*, Junmei Feng, Jinchang Ren , *Senior Member, IEEE*, and Heng Luo

***Abstract*— Drone-view object detection (DroneDet) models typically suffer a significant performance drop when applied to nighttime scenes. Existing solutions attempt to employ an exposure-adjustment module to reveal objects hidden in dark regions before detection. However, most exposure-adjustment models are only optimized for human perception, where the exposure-adjusted images may not necessarily enhance recognition. To tackle this issue, we propose a novel Detection-driven Exposure-correction network for nighttime DroneDet, called DEDet. The DEDet conducts adaptive, nonlinear adjustment of pixel values in a spatially fine-grained manner to generate DroneDet-friendly images. Specifically, we develop a fine-grained parameter predictor (FPP) to estimate pixelwise parameter maps of the image filters. These filters, along with the estimated parameters, are used to adjust pixel values of the low-light image based on nonuniform illuminations in drone-captured images. In order to learn the nonlinear transformation from the original nighttime images to their DroneDet-friendly counterparts, we propose a progressive filtering module that applies recursive filters to iteratively refine the exposed image. Furthermore, to evaluate the performance of the proposed DEDet, we have built a dataset NightDrone to address the scarcity of the datasets specifically tailored for this purpose. Extensive experiments conducted on four nighttime datasets show that DEDet achieves a superior accuracy compared with the state-of-the-art (SOTA) methods. Furthermore, ablation studies and visualizations demonstrate the validity and interpretability of our approach. Our NightDrone dataset can be downloaded from https://github.com/yuexiemail/NightDrone-Dataset.**

***Index Terms*— Adverse illumination conditions, differentiable image filters, drone-view object detection (DroneDet), exposure correction.**

Manuscript received 24 July 2023; revised 30 October 2023 and 2 December 2023; accepted 30 December 2023. This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant XJSJ23104 and Grant XJSJ23101 and in part by Guangzhou Basic and Applied Basic Research Foundation under Grant 2023A04J1742. *(Corresponding authors: Qiguang Miao; Jinchang Ren.)*

Yue Xi, Junmei Feng, and Heng Luo are with the Guangzhou Institute of Technology, Xidian University, Guangzhou, Guangdong 510555, China (e-mail: xiyue@xidian.edu.cn; fengjunmei@xidian.edu.cn; 23031212076@stu.xidian.edu.cn).

Wenjing Jia is with the Global Big Data Technologies Centre and the Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, NSW 2007, Australia (e-mail: Wenjing.Jia@uts.edu.au).

Qiguang Miao is with the Xi'an Key Laboratory of Big Data and Intelligent Vision and the School of Computer Science and Technology, Xidian University, Xi'an, Shaanxi 710071, China (e-mail: qgmiao@xidian.edu.cn).

Jinchang Ren is with the National Subsea Centre, Robert Gordon University, AB10 7AQ Aberdeen, U.K. (e-mail: jinchang.ren@ieee.org).

I. Introduction

DRONE-VIEW object detection (DroneDet) is a drone-vision technique for locating objects and predicting their categories in images captured by drones. It has gathered great momentum in various drone-vision applications [1], including aerial object tracking [2], aerial person reidentification [3], and aerial crowd counting, among others. Despite the remarkable advances in DroneDet [4], [5], [6], [7] in recent years, object detection at night remains a challenging because of adverse illumination conditions at night. For instance, a widely used detector YOLOv5 [8] achieves an average precision (AP) at an intersection-over-union (IoU) threshold of 0.5 for all classes (AP_{50}) of only 41.2% on the VisDrone (Night) dataset,[1] in contrast to the AP_{50} of 67.3% achieved on the Microsoft COCO [10]. This performance drop of nearly 30% is largely attributed to the presence of small objects in adverse illumination conditions.

Images captured under adverse illumination conditions usually contain underexposure and/or overexposure, significantly degrading the image quality and the performance of following-on analysis. Underexposure errors result in very dark regions, which suffer from low signal-to-noise ratios (SNRs) and contrast due to limited photons, whereas overexposure errors result in very bright and saturated image regions due to the limited range of camera sensors. These adverse illumination conditions severely restrict the availability of valuable information necessary for robust object detection.

One intuitive solution to address the adverse illumination issue is to first generate well-exposed images using an image enhancement module. These well-exposed images are then fed into a detection module for object detection. However, the performance improvement brought by this separate fashion is very limited because the primary goal of the enhancement module is to improve visual or perceptual quality for human viewing rather than specifically optimizing the task of DroneDet. Another solution involves cascading the enhancement module and detection module to optimize both of them using enhancement or detection losses. Although this solution aligns with logic, it fails to achieve satisfactory detection performance. The existing enhancement modules typically perform image-to-image translation with neural networks [11]. As a result,

[1] VisDrone (Night) includes only nighttime images and their corresponding labels carefully selected from the VisDrone datasets [9].

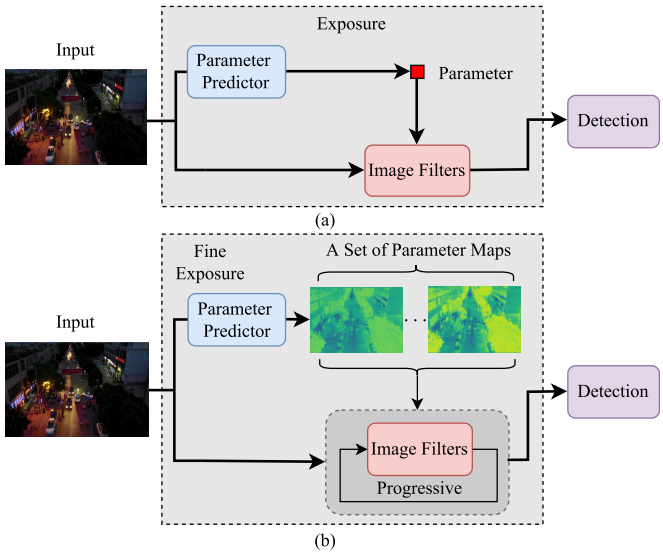


Fig. 1. Comparisons of different pipelines. (a) Pipeline of IA-YOLO [12]. It follows a rigid exposure paradigm, which applies an image filter with the same parameter on every pixel in an image. (b) Our DEDet estimates a set of pixelwise parameter maps and then progressively applies filters with the estimated parameters on an image to cope with the issue of nonuniform illumination in a drone-captured image.

these translated images may contain a large number of artifacts that can be harmful to the subsequent detection module.

Recently, a novel model called IA-YOLO [12] has emerged. As shown in Fig. 1(a), instead of direct image-to-image translation, a differentiable image filter was designed to enhance the nighttime images for a detection module, with parameters estimated by a neural network. In practice, there are substantial illumination variations in different regions of large scenes in drone-captured images. Nighttime, drone-view images covering large-scale scenes are typically captured under different types of lights, such as streetlights, headlights and taillights of a vehicle, and lights of stores along the street. It is unrealistic to expect an image filter with a fixed parameter to effectively process every pixel exposed under different lighting conditions. This leads to the question "How to design a fine exposure module capable of correcting every pixel value based on the distribution of illuminations in a nighttime image," so as to improve nighttime DroneDet performance by deploying more accurate estimations.

To answer this question, we propose Detection-driven Exposure-correction network for nighttime DroneDet (DEDet), an exposure-based detection network specifically designed for nighttime DroneDet. Our DEDet improves the DroneDet performance in low-light conditions by applying a novel technique, "detection-driven exposure-correction," on nighttime images. As illustrated in Fig. 1(b), DEDet consists of an exposure-correction module and a detection module, jointly optimized in an end-to-end training paradigm. The exposure-correction module generates nighttime DroneDet-friendly images through the cooperation of the fine-grained parameter predictor (FPP) module and the progressive filtering module (PFM). Specifically, the FPP estimates pixelwise exposure parameter maps of a series of image filters using a transformer-based neural network. The PFM conducts a complex nonlinear transformation to each pixel value by iteratively applying the image filters estimated by FPP. The detection module then conducts DroneDet on the exposed images generated by the exposure module. Last but not least, to facilitate the evaluation of DEDet and address the scarcity of datasets for nighttime DroneDet, we collected real-world images using a drone and created a new drone-captured dataset, called NightDrone.

The key contributions made in this study are as follows.

1) We propose a DEDet framework to expand DroneDet from daytime scenes to nighttime scenes by learning fine-grained exposure maps and obtain nighttime DroneDet-friendly images.
2) We develop the FPP and PFM modules that can cooperate together to obtain nighttime DroneDet-friendly images, leading to further improved DroneDet performance.
3) We create the NightDrone dataset, the first drone-captured dataset for nighttime DroneDet, to advance the capabilities of DroneDet in adverse illumination conditions.

The remainder of this study is organized as follows. Section II provides a comprehensive review of nighttime DroneDet and summarizes related works. Section III presents details of our DEDet. Section IV introduces the new NightDrone dataset created for nighttime DroneDet. Section V presents the experimental results of DEDet. Section VI concludes this article and points out future research directions.

II. Related Work

Nighttime DroneDet is closely related to three key techniques, i.e., DroneDet, low-light image enhancement (referred to as "LLIE"), and low-light object detection (referred to as "LLDet"). In this section, we briefly summarize the recent developments in these three techniques.

A. DroneDet: Drone-View Object Detection

DroneDet [9] is an emerging research topic in remote sensing. Images captured by drones contain numerous small objects, which can significantly degrade detection performance. Existing methods can be classified into three groups: super-resolution-based (SR-based) methods, context-based methods, and representation fusion-based (RF-based) methods.

SR-based methods [6], [13], [14] enhance low-resolution regions of interest (RoIs) into high-resolution ones using super-resolution techniques. These models generally consist of candidate proposal, super resolution, and detection modules, regarded as a multistage strategy. However, the multistage strategy can be inefficient and challenging to train effectively.

Context-based methods [15], [16], [17] leverage local context of surrounding objects and global context in an image, and integrate both of these contexts into the original objects' features. Building such contextual relationships proves challenging due to the complexity and diversity of backgrounds in drone-captured images.

RF-based methods [18], [19], [20], [21] combine detailed spatial information in shallow representations and semantic

information in deep representations for improving aerial object detection. Dong et al. [18] proposed a novel attention-based multilevel feature fusion module to adaptively fuse multilevel features and generate more powerful pyramidal features for object detection in aerial images. Ye et al. [19] proposed an adaptive attention fusion mechanism for fusing features to boost the representation power.

Although these methods have obtained impressive DroneDet performance, most of them fail to recognize objects at night. Some recent works [22], [23] have started to fuse infrared and RGB images to improve the detection performance. Infrared images provide complementary information for RGB images due to their high-sensitivity in night vision. However, infrared cameras are not equipped in most commercial drones, mainly due to the limitations of the manufacturing process and the cost concerns. Therefore, our focus is to improve the DroneDet on RGB images under poor visibility conditions caused by adverse illumination conditions.

B. LLIE: Low-Light Image Enhancement

LLIE reveals information hidden in the dark areas of an image to improve the image's quality for human perception. Early methods for LLIE adopted intensity mapping and local statistics, including exposure correction [24] and histogram equalization [25]. Later, Retinex-based methods [26], [27], [28], [29] disentangled illumination-related and reflectance-related components from low-light images by introducing certain priors based on empirical observations. Recently, neural-network-based methods [30], [31], [32], [33] have shown impressive results in LLIE. The pioneering work LLNet [30] utilized stacked sparse deep autoencoders to brighten and denoise low-light images. KinD [31] was proposed to kindle the darkness, which decomposes low-light images into an illumination component for light adjustment and a reflectance component for degradation removal. Xu et al. [32] proposed an SNR-aware network, which exploited the long-range operation for global context and the short-range operation for local context for pixelwise enhancement in a spatial-varying manner. Yang et al. [33] proposed the sparse gradient minimization subnetwork (SGM-Net) and built a coupled representation with l_0 gradient minimization.

When these LLIE methods are used for nighttime DroneDet, there is still room for further improvement. After all, these methods are designed for human perception rather than machine vision. In this article, we propose a detection-driven exposure-correction mechanism to strengthen the weak collaboration between LLIE and DroneDet.

C. LLDet: Low-Light Object Detection

Few works have paid attention to the improvement of detection performance after low-light enhancement [34]. Detectors used for LLDet are categorized into three categories.

1) *Directly Trained Detectors:* The models in this category are trained directly on nighttime images, as shown in Fig. 2(a).

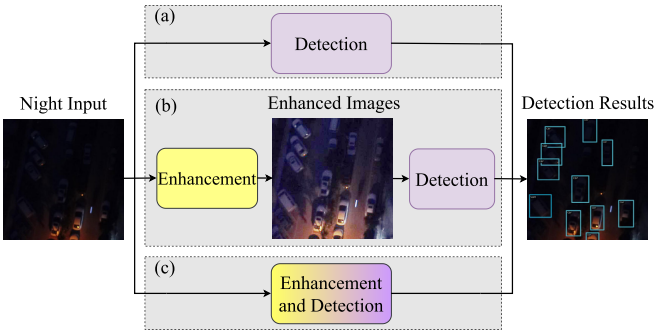


Fig. 2. Comparison of detection methods with an enhancement module under adverse illumination conditions. (a) Directly trained detectors. (b) Separately trained detector. (c) Jointly trained detectors.

2) *Separately Trained Detectors:* An enhancement module is first trained to brighten nighttime images, and then, a pretrained detection module is trained on the enhanced images, as shown in Fig. 2(b).
3) *Jointly Trained Detectors:* The enhancement and detection modules are jointly trained in an end-to-end manner on nighttime images, as shown in Fig. 2(c).

It is hard for directly trained detectors [35], [36], [37], [38], [39] to learn sufficiently discriminative features for detection due to low SNRs and contrast brought by image degradation. Yim and Sohn [35] proposed a dual-channel convolutional architecture to deal with quality degradation under low-light conditions. For separately trained detectors [40], enhancement and detection modules are optimized separately, which is also referred to as a two-stage strategy. However, it is problematic to evaluate the connection between the two modules. Most methods merely employed object detection performance as an evaluation metric of enhancement modules. Although the two-stage strategy may boost the detection performance, they often lack robustness and generalization. Pei et al. [40] suggested that enhancement methods sometimes fail to improve detection performance and can even lead to a reduction in detection performance.

Recently, few works have focused on the third category, the jointly trained detectors [11], [12], [41], [42], [43]. They started to explore the connection between image exposure and DroneDet. Ma et al. [42] proposed an illumination allocator following a parallel architecture to deal with LLDet. In the architecture, the latent representations between image exposure and detection are built to mutually enhance each other. Xue et al. [43] proposed a cascaded architecture for low-light exposure and object detection. Following the cascaded architecture, the recurrent exposure generation network (REGNet) [11] simulated the nonlinear process of the multiexposure technique by generating a series of pseudo-exposure images. Instead of image-to-image mapping for image exposure, Liu et al. [12] reformatted the exposure task as parameters of image filters to enhance low-light images for better object detection. Different from the aforementioned methods, we propose to learn the pixelwise parameter maps of image filters to adjust each pixel value in a fine exposure-correction manner to deal with the nonuniform illumination in drone-captured images.

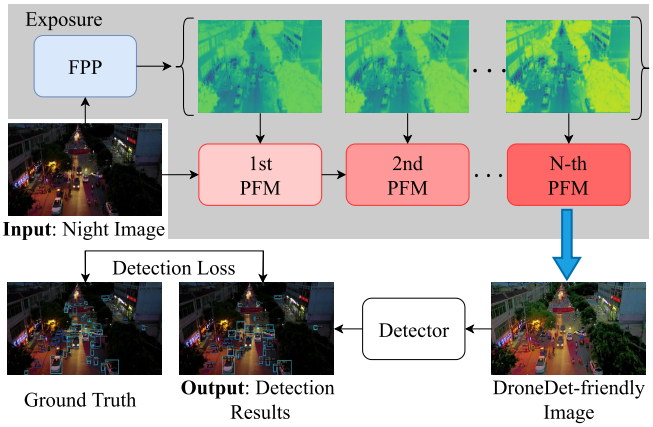


Fig. 3. Pipeline of our DEDet, which contains three modules. The first module FPP is responsible for estimating the parameter maps of image filters for a night image. The second module PFM uses the image filters with the estimated parameters to adjust the image exposure. The last module Detector performs the final object detection on the DroneDet-friendly image. The three modules are optimized together in an end-to-end training fashion.

III. Proposed Method

A. Overview

The proposed DEDet employs a detection-driven exposure-correction mechanism to generate nighttime DroneDet-friendly images so as to improve the accuracy of nighttime DroneDet. Fig. 3 illustrates the workflow of DEDet, which consists of three key modules: FPP, PFM, and Detector. The FPP module estimates multiple pixelwise parameter maps for an input image, where each value corresponds to individual pixels within the image. The PFM module then adjusts the image's exposure using the parameters estimated by FPP to generate an exposure-corrected image that produces better DroneDet performance. Note that the PFM module is iteratively deployed for multiple times with varying parameters to progressively refine the enhanced images. Finally, the Detector module locates and classifies objects of interest in the images.

B. FPP: Fine-Grained Parameter Predictor

Existing filters with adaptive parameters often assume a uniform illumination condition for the entire image. However, in real-world scenarios, illumination conditions vary spatially in different regions of the drone-captured images due to the large scene these images encompass, typically captured under various lighting conditions.

To overcome these limitations, we propose the FPP module, which estimates pixelwise parameter maps to adjust each pixel value. As illustrated in Fig. 4, the FPP module contains three submodules: the residual feature encoder, the latent feature extractor, and the residual feature decoder.

1) Residual Feature Encoder: An input image is defined as $I \in \mathbb{R}^{W\times H\times C_{\text{in}}}$, where W, H, and C_{in} are the width, height, and channel number of the image, respectively. We use two residual convolutional blocks (RCBs) $\mathcal{E}(\cdot)$ to extract the feature $F \in \mathbb{R}^{(W/4)\times(H/4)\times C}$ as

$$F = \mathcal{E}(I) \tag{1}$$

where C is the number of the feature channels. The convolutional layers are beneficial to visual processing, allowing transformers in the latent feature extractor to converge more stably and rapidly [44]. In addition, to improve the encoding efficiency, F is obtained with two max-pooling layers by downsampling the original image I.

2) Latent Feature Extractor: After the input image is mapped into a higher dimension latent space with the residual feature encoder, we use S Swin transformer blocks (STBs), denoted as $\mathcal{H}(\cdot)$, to extract the features $F' \in \mathbb{R}^{(W/4)\times(H/4)\times C}$ in the latent space from F as

$$F' = \mathcal{H}(F). \tag{2}$$

In particular, the intermediate features $\{F_0, F_1, \ldots, F_{S-1}\}$ and the output feature F' are extracted by a series of cascaded STB as

$$F_{i+1} = \mathcal{H}_i(F_i), \quad i = 0, 1, 2, \ldots, S-1 \tag{3}$$

where $\mathcal{H}_i(\cdot)$ represents the ith STB, $F_0 = F$, and $F' = F_S$.

As illustrated in Fig. 4, each STB consists of a standard multihead self-attention (MSA) module, a two-layer multilayer perception (MLP) module, and a residual connection applied after each module. The MSA allows the model to extract jointly to information in different feature subspaces, which is beneficial for dealing with nonuniform illumination in the image. The MLP transforms the input to a higher dimensional space and then restores to the original dimension as the input.

3) Residual Feature Decoder: The residual feature decoder $\mathcal{D}(\cdot)$ is used to estimate the parameter maps of image filters. Initially, two upsampling residual convolutional blocks (URCBs) are utilized to upsample the latent feature F' back to the original input size. Then, a convolutional layer is employed to generate N parameter maps, denoted as $\mathbf{P} \in \mathbb{R}^{W\times H\times C'}$, as

$$\mathbf{P} = \mathcal{D}(F') \tag{4}$$

where $C' = N \times N_f$ is the channel number of the parameter maps, and N_f is the number of image filters.

C. PFM: Progressive Filtering Module

1) Progressive Filtering: Experienced engineers often employ various image filters, such as Gamma correction and image sharpening, to create more expressive and visually appealing images. However, manually tuning the hyperparameters of these image filters for a broad range of scenes based on visual perception and experience can be very time-consuming.

We employ a progressive scheme to adjust the image exposure, which first generates a coarse image using the filters with the first parameter map and then gradually refines the coarse image using filters with other parameter maps. Specifically, given a pixel value P_{in} in I and a pixel value P_{out} in a DroneDet-friendly image, the pixel value adjustment from P_{in} to P_{out} is a complex nonlinear mapping. To enable more versatile adjustments, we progressively apply image filters with different parameter maps as

$$P_{\text{in}}^{i+1} = \mathcal{F}_i(P_{\text{in}}^i), \quad 0 \le i \le N-1 \tag{5}$$

where N denotes the number of iterations, $P_{\text{in}}^0 = P_{\text{in}}$, and $P_{\text{out}} = P_{\text{in}}^N$. $\mathcal{F}_i(\cdot) = \mathcal{G}((\cdot)^\gamma, \lambda)$, $\mathcal{G}(\cdot, \lambda)$ is a sharpening filter, and $(\cdot)^\gamma$ is a Gamma function used to control image exposure

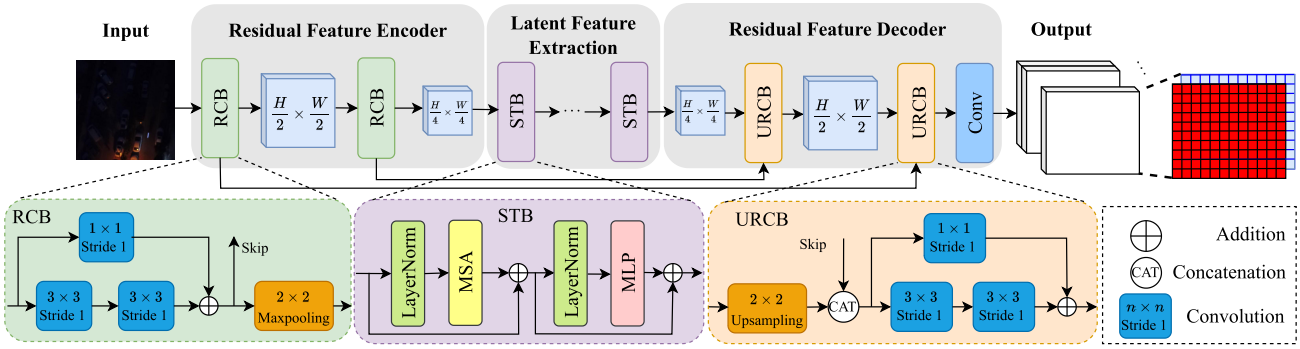


Fig. 4. Details of FPP, which is composed of three modules: the residual feature encoder, the latent feature extractor, and the residual feature decoder.

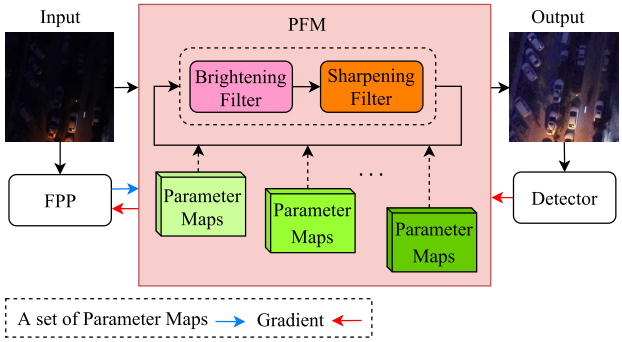


Fig. 5. Details of PFM, which consists of the brightening filter for image exposure and sharpening filter for image sharpening. The differentiability of PFM allows the training of neural networks by backpropagation.

levels. We set $N = 4$, as a tradeoff by taking into consideration of the computational cost and performance gain.

In addition, to ensure the effectiveness of the gradient-based optimization of DEDet, the design of PFM needs to adhere to the principle of differentiability. PFM needs to be differentiable in terms of their filters' parameters for gradient-based optimization of DEDet. It is necessary to ensure that each module in a neural network is differentiable in terms of its parameters to train the neural network with gradient backpropagation. As shown in Fig. 5, the PFM module cascades two differentiable image filters: a brightening filter and a sharpening filter.

2) Brightening Filter: The brightening filter brightens a low-light image by mapping pixel values P_n of the nighttime image to pixel values P_{br} of the brightened image with power transformations. The brightening filter is described as

$$P_{\text{br}} = P_n^{\gamma} \tag{6}$$

where γ is the brightening filter's parameter, which is differentiable on both P_n and γ.

3) Sharpening Filter: We use the Gaussian filter Gauss$(\cdot)$ for a better detection performance by adding the details of an image. The image sharpening process is described as follows:

$$\mathcal{G}(P_{\text{br}}, \lambda) = P_{\text{br}} + \lambda(P_{\text{br}} - \text{Gauss}(P_{\text{br}})) \tag{7}$$

where λ is a positive scaling factor, and the higher the λ is, the more the details are added to the image. Moreover, $\mathcal{G}(\cdot, \lambda)$ is differentiable with respect to P_{br} and λ. The degree of image sharpening is tuned by optimizing λ to obtain a better detection performance. As for the Gaussian filter Gauss$(\cdot)$, we follow the practice described in this article [12] and set its σ as 5 and its kernel size as 7. The parameters of the kernels are sampled from the 2-D Gaussian function and constant during the training process.

D. Unsupervised Pretraining for the Exposure Module

Most exposure networks rely on paired images of low-light and normal-light images for supervised training [45]. These paired images are exhaustively collected through illumination condition adjustments, modifying cameras' parameter settings during data collection, or retouched by experts to enhance the visual appearance of images.

Nonetheless, it is labor-intensive to collect numerous paired drone-captured images for training our exposure module. To overcome the limitation, we propose an unsupervised pretraining strategy for image exposure, where no paired or unpaired data are required during the training process. In addition, the unsupervised pretraining strategy is beneficial to accelerate the joint training process and boost the DroneDet accuracy. Two differentiable losses are adopted to evaluate the quality of exposed images for pretraining our exposure module. We define the total loss $\mathcal{L}_t$ as follows:

$$\mathcal{L}_t = \mathcal{L}_s + \mathcal{L}_e \tag{8}$$

where $\mathcal{L}_s$ and $\mathcal{L}_e$ are the smoothness and exposure losses, respectively.

1) Smoothness Loss: According to the total variation loss [46] for image reconstruction, a smoothness loss is designed to the parameter maps $\mathbf{P}$ to preserve the smoothness property and structural information of illumination conditions. L_s is described as follows:

$$\mathcal{L}_s = \frac{1}{N}\sum_{i=1}^{N} ||\nabla_v \mathbf{P}_i||_1 + ||\nabla_h \mathbf{P}_i||_1 \tag{9}$$

where N is the same as the iteration number N in (5), ∇_v and ∇_h are the vertical and horizontal gradient operations of $\mathbf{P}$, respectively, and $||\cdot||_1$ is the the $L1$-norm.

2) Exposure Loss: To mitigate underexposed or overexposed areas, an exposure loss $\mathcal{L}_e$ is designed to control the exposure level of every local region in an exposed image. The normalized intensity of each pixel, ranging from 0 (underexposed) to 1 (overexposed) in the exposed image, is required to retreat from 0 or 1. $\mathcal{L}_e$ is utilized to calculate the distance between the average intensity value of a local region and the

well-exposedness level E. According to exposure fusion [47], E is set to 0.6 in the experiments. $\mathcal{L}_e$ is described as follows:

$$\mathcal{L}_e = \frac{1}{K}\sum_{i=1}^{K}|A_i - E| \tag{10}$$

where K is the number of nonoverlapping local regions R, whose sizes are 32×32 pixels, and A_i is the average intensity of each R_i in an exposed image. By minimizing $\mathcal{L}_e$, we can optimize the enhancement process to achieve desired exposure levels across the image.

E. Backbone Detector

We employ the detector YOLOv5 [8] as the detector module of DEDet. YOLOv5 is a popular detector and widely deployed in numerous real-world scenarios, including video surveillance and autonomous driving. Compared with its previous versions, YOLOv5 introduces Mosaic data augmentation and adaptive anchor computation for efficient training. In addition, YOLOv5 incorporates the focus layers and the CSP [48] structure into the original Darknet backbone for robust feature extraction, which improves the small object detection accuracy. In this work, the same network architecture and loss functions as YOLOv5 are utilized for training DEDet.

IV. NightDrone Datasets

Currently, there is a lack of benchmark datasets specifically collected for nighttime DroneDet. As stated in [9] on drone vision, two drone-captured benchmark datasets VisDrone and DroneVehicle are widely used for DroneDet. Moreover, only a small portion of images in both of the drone-based datasets is nighttime images, and the majority of the images in these two datasets were collected in the daytime. To fill the gap in available datasets for nighttime DroneDet, we have created the NightDrone dataset. In this section, we first present the data collection and annotation process of this dataset, and then compare it with the existing DroneDet datasets.

A. Nighttime Drone-Captured Image Collection

We used a DJI MINI 2 drone equipped with a camera to capture image data at nighttime for this drone dataset "NightDrone." This dataset encompasses a variety of aerial scenes, such as urban and rural streets, pedestrian malls, and parking lots. We deliberately conducted data collection at different times of the day, including evening, dusk, and midnight, to increase the diversity in images' illumination. In addition, during data collection, we adjusted the drone's height and shooting angle to simulate different scenarios in the real world. To increase the diversity of objects' scales, the drone used for data collection is controlled to fly between 60 and 120 m. Similarly, to increase the diversity of objects' appearance, the drone is flying with three shooting angles (30°, 60°, and 90°). The above settings make DroneDet on our NightDrone dataset even more challenging.

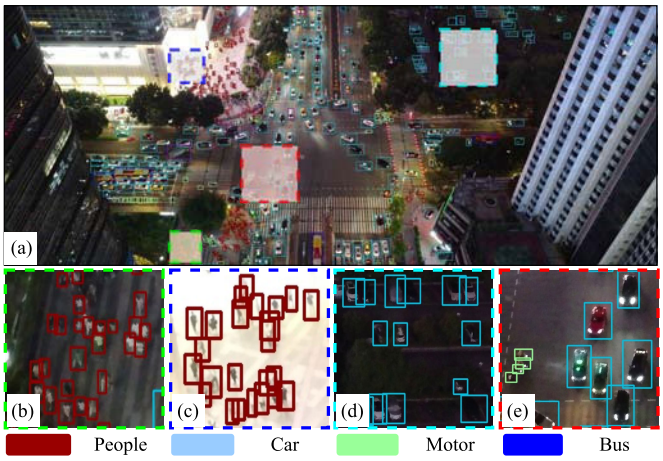


Fig. 6. Example from our NightDrone dataset. (a) Typical image including numerous instances under adverse illumination conditions. (b) Illustration of "People" instances under a low-light region. (c) Illustration of "People" instances under an overexposed region. (d) Illustration of "Car" instances under a low-light region. (e) Illustration of "Car" and "Motor" instances under a normal-light region. Four out of eight of the possible categories in NightDrone are shown.

B. Nighttime Image Annotation

We annotate objects of interest in drone-captured images with rectangular bounding boxes (BBoxes). A BBox is defined as $[(x_c, y_c), w, h, c]$, where (x_c, y_c), w, h, and c are its center coordinate, its width and height, and the object's category, respectively. The annotation is done using the labeling software "LabelImg."[2] Finally, the annotations of every image are saved as an XML file following the PASCAL VOC [49] format.

It is worth noting that annotating images captured by drones at night is very challenging due to the presence of numerous small-size objects in poor illumination conditions, as shown in Fig. 6. To ensure the precision and accuracy of the annotation, instead of the original nighttime images, we annotate the reconstructed nighttime images generated by low-light enhancement algorithms. The self-calibrated illumination (SCI) model [50] is adapted to brighten these low-light images to reduce the probability of incorrect or missing labels during manual annotation.

C. Statistics for Our NightDrone Dataset

Our NightDrone Dataset includes 6805 images collected by drones. This dataset includes 5445 training images and 1360 testing images. The 254 222 objects were manually annotated with BBoxes. Each image in NightDrone includes an average of 37.36 objects, of which the maximum number is 673. Fig. 7(a) shows the histogram of the number of annotated objects per image. Additionally, we define eight common object categories in drone application: Car, People, Motor, Van, Bicycle, Truck, Bus, and Tricycle. Different from the ten object categories defined in VisDrone, we do not distinguish People and Pedestrian, and Tricycle and Awning-tricycle due to their similar appearance at night. The pie chart in Fig. 7(b) shows the proportion of objects of the predefined

[2]The software is available at https://github.com/heartexlabs/labelImg.

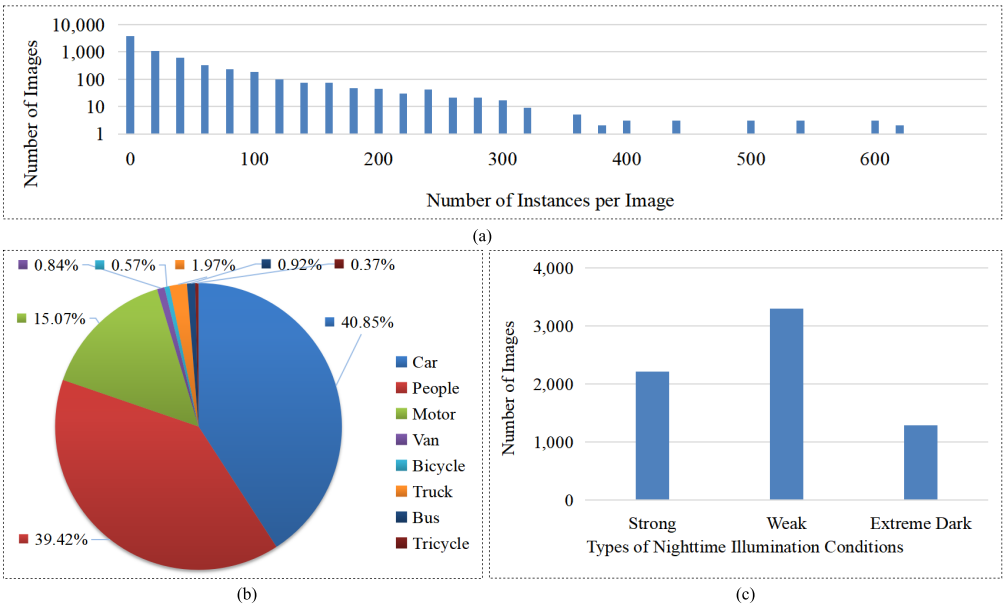
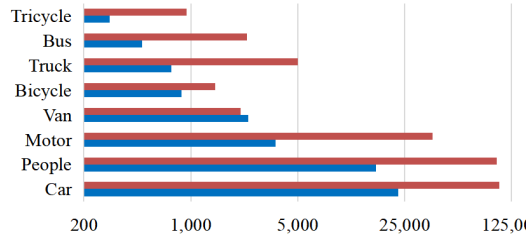


Fig. 7. Statistics of our NightDrone dataset. (a) Histogram of the number of annotated instances per image. (b) Proportion of objects of different categories. (c) Histogram of the number of images in three types of nighttime illumination conditions: "Strong," "Weak," and "Extreme Dark."



	Car	People	Motor	Van	Bicycle	Truck	Bus	Tricycle
NightDrone	103,421	99,795	38,158	2,117	1,444	4,994	2,323	936
VisDrone (Night)	22,581	16,197	3,586	2,378	865	744	479	296

Fig. 8. Comparison of the numbers of instances in each category between our NightDrone and VisDrone (Night).

eight categories. Its unbalanced distribution might result in a performance drop of DroneDet.

Following ExDark [51], all the images in NighDrone are grouped into three types of nighttime illumination conditions: "Strong," "Weak," and "Extreme Dark." The "Strong" type refers to images with multiple visible and relatively bright light sources, the "Weak" type refers to images with a few visible and weak light sources, and the "Extreme Dark" type refers to images with very low illumination. Fig. 7(c) presents the histogram of the number of images in the three types of nighttime illumination conditions.

D. Comparison With Existing Drone-Captured Datasets

We compare our NightDrone with the two existing drone-captured image datasets, namely, the VisDrone and DroneVehicle dataset. VisDrone [9] is a benchmark dataset collected using a drone platform DJI under diverse lighting conditions and annotated with ground truth for DroneDet. However, most of the images in VisDrone were captured during the daytime with good illumination conditions. Nighttime images account for 17.9% of the whole dataset. Fig. 8 shows the comparison of the number of instances in each category between our NightDrone and VisDrone (Night).

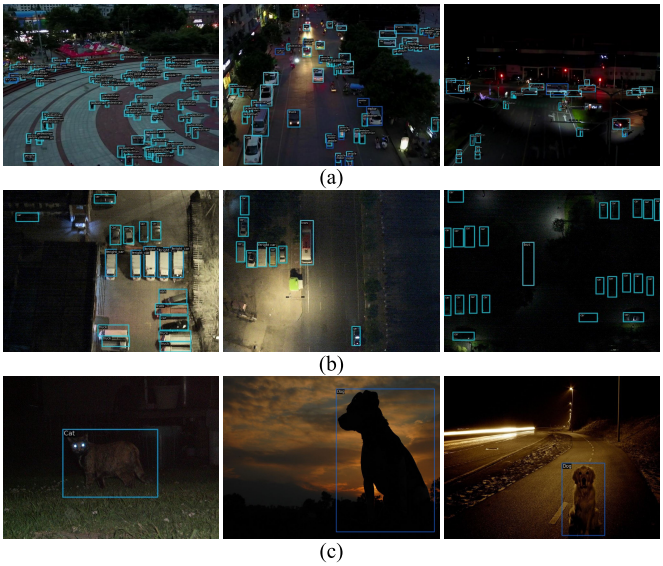


Fig. 9. Examples from the experimental datasets. (a) Images from our VisDrone (Night) dataset. (b) Images from our DroneVehicle (Night) dataset. (c) Images from the ExDark dataset.

The DroneVehicle dataset [22] is a drone-based cross-modality dataset. It includes pairs of RGB and infrared images covering a wide range of scenarios from daytime to nighttime. However, the annotated objects in this dataset are mainly larger-sized objects, such as trucks and cars, while small-sized objects, such as pedestrians and motors, are not annotated.

In contrast, all images in the NightDrone dataset were collected at nighttime under poor lighting conditions. Furthermore, the annotated objects in NightDrone include both large-sized and small-sized objects.

V. Experiments

We conducted extensive experiments on four datasets and compared the detection performance of DEDet with seven state-of-the-art (SOTA) methods to validate its effectiveness

TABLE I
OVERVIEW OF THE EXPERIMENTAL DATASETS

Dataset	#BBox	#Image	#Category	Year
VisDrone (Night)	47,426	1,261	10	2021
DroneVehicle (Night)	86,435	4,717	5	2022
NightDrone	254,222	6,805	8	2023
ExDark	23,710	7,363	12	2019

for nighttime DroneDet. Next, we first introduce the experimental datasets and then present the detailed experimental results.

A. Datasets

The real nighttime DroneDet performance is evaluated on not only our NightDrone dataset but also the nighttime images of the two drone-captured datasets, VisDrone and DroneVehicle, which are referred to as "VisDrone (Night)" and "DroneVehicle (Night)," respectively. In addition, we also use a general object dataset called ExDark [51]. These four datasets encompass a diverse range of scenarios and capturing conditions. Table I presents a summary of these four datasets.

1) VisDrone (Night): From VisDrone [9], we carefully selected the nighttime images and their corresponding annotation files to build our experimental dataset VisDrone (Night),[3] which includes 1008 training images and 253 testing images. The spatial resolution of the images is approximately 2000 × 1500 pixels. These images are annotated with giving BBoxes into ten predefined categories, i.e., Truck, Bus, Car, Van, Awning-tricycle, Tricycle, Motor, Bicycle, Person, and Pedestrian, the same as those in VisDrone, as shown in Fig. 9(a).

2) DroneVehicle (Night): As the first drone-based cross-modality dataset for vehicle detection, DroneVehicle [22] contains pairs of RGB and infrared images. Similarly, we carefully selected image pairs collected during the nighttime and then chose only RGB images with annotated ground truth to build the DroneVehicle (Night)[3] dataset. As a result, the new DroneVehicle (Night) includes 4279 training images and 438 testing images, where each image has a spatial resolution of 640 × 512 pixels. The images are annotated with defined BBoxes into five vehicle categories, i.e., Bus, Fright Car, Truck, Car, and Van, the same as those in DroneVehicle, as shown in Fig. 9(b).

3) ExDark: ExDark is a natural scene dataset widely used for LLDet. It comprises a total of 7363 images, which are annotated with BBoxes into 12 predefined categories, namely, Bus, Car, Bicycle, Boat, Motorbike, Table, Chair, Cup, Bottle, People, Cat, and Dog. The 80% of these images from each category are used for training and the remaining 20% for testing. The resolution of these images is approximately 640 × 480 pixels, as shown in Fig. 9(c).

B. Implementation and Evaluation Metrics

1) Implementation Details: We implemented the proposed DEDet based on PyTorch 1.8.1, and all models were trained using one computing node of two NVIDIA RTX3090 GPU cards each with 24-GB memory. The pretrained weights[4] were utilized to expedite the training process. The Adam optimizer was used for training because it dealt with sparse gradients in backpropagation more effectively and converged faster than the stochastic gradient descent (SGD) optimizer. We adopted the cosine learning rate schedule with a learning rate of $3e^{-4}$, and set the batch size to 8. Additionally, several data augmentation techniques were utilized, including Mosaic, Random Affine, and hue value saturation (HSV) random augmentation. Considering the limited computing resources onboard, the input images in our experiments were all resized to 640 × 640 pixels without any tricks, such as multiscale testing, while larger input images could actually improve the detection accuracy of the DroneDet.

2) Evaluation Metrics: Similar to the PASCAL VOC Benchmark [49], the detection performance is evaluated by using the metric called AP across IoU with different thresholds. To be specific, AP is averaged over ten IoU thresholds, which range from 0.50 to 0.95 with a fixed interval of 0.05. AP_{50} and AP_{75} are calculated on the single IoU thresholds of 0.5 and 0.75, respectively. AP is calculated by the integral of the Precision–Recall curve $p(r)$, given by

$$\mathrm{AP} = \int_{r=0}^{1} p(r)dr \tag{11}$$

where p is the Precision, and r is the Recall

$$p = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{12}$$

$$r = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{13}$$

where TP is the number of correctly predicted positive instances; FP is the number of incorrectly predicted positive instances; and FN is the number of missing positive instances during the detection.

C. Ablation Studies

To verify the effectiveness of each component in DEDet, we conducted ablation studies on our NightDrone dataset, using YOLOv5 [8] as the baseline.

1) Effectiveness of Image Filters in PFM: We conduct quantitative AP evaluation on the brightening filter and sharpening filter to verify the effectiveness of image filters adopted in PFM. We first remove the brightening filter while retaining the sharpening filter, referred to as "DEDet w/ Sharping." Then, we remove the sharpening filter while retaining the brightening filter, referred to as "DEDet w/ Brightening." The model with both filters is referred to as "DEDet w/ Sharping & Brightening." Table II shows that brightening filter and sharpening filter improves AP_{50} accuracy by 2.7% and 3.1%, respectively, compared with the baseline model. Moreover, combining the two filters achieves the highest detection accuracy, showing the effectiveness of these filters.

[3]The VisDrone (Night) and DroneVehicle (Night) dataset can be freely accessed from the link: https://github.com/yuexiemail/NightDrone-Dataset.

[4]The pretrained weights of YOLOv5 [8] can be downloaded from this URL https://download.openmmlab.com/mmyolo/v0/yolov5.

TABLE II

RESULTS OF ABLATION STUDIES ON DIFFERENT IMAGE FILTERS IN PFM

Method	Filters		FPP w/ Pre-training	AP_{50}[%]
	Brightening	Sharpening		
Baseline	×	×	×	58.3
DEDet w/ Sharping	×	✓	×	61.0
DEDet w/ Brightening	✓	×	×	61.4
DEDet w/ Sharping & Brightening	✓	✓	×	62.7
DEDet w/ Sharping & Brightening	✓	✓	✓	64.1

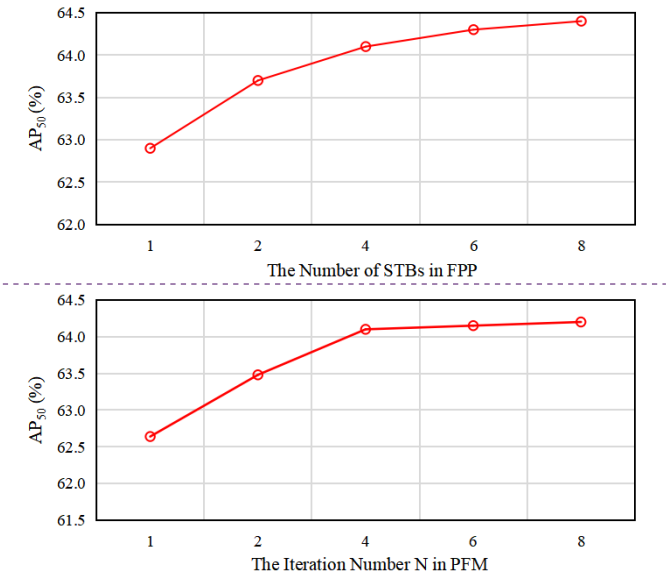


Fig. 10. Ablation studies on different STB Numbers in FPP and iteration numbers N in PFM.

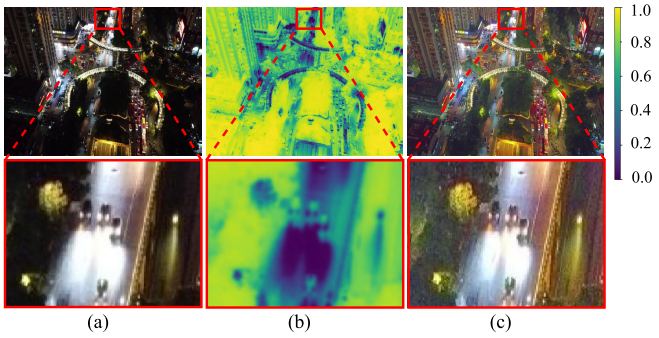


Fig. 11. Visualization of the parameter maps estimated by the FPP. (a) Original nighttime images. (b) Parameter maps of the brightening filter in PFM. (c) Exposed images after applying the estimated parameters.

2) Impact of the STB Number in FPP and Iteration Number in PFM: Fig. 10 presents the effects of the STB number in FPP and the iteration number N in PFM on the DEDet detection accuracy AP_{50}, respectively. It is observed that the AP_{50} is positively correlated with both hyperparameters. As the STB number and iteration number increase, the performance gain tends to be saturated gradually, and the computational cost increases. Therefore, taking both the detection performance gain and computational cost into consideration, setting both of the STB number and the iteration number N to be 4 is optimal in terms obtaining a relatively lightweight model.

3) Visualization of the Parameter Maps Generated by FPP: We visualize pixelwise parameter maps to demonstrate the effectiveness of the FPP in adjusting the parameters of filters in the PFM module based on the illumination conditions in the images. In Fig. 11, images in (a) are the original input images and (b) shows the averaged parameter maps of the brightening filter of all iterations, which are then normalized to the range of [0, 1]. Finally, Fig. 11(c) illustrates the images exposed with the parameter maps. The visualization in Fig. 11(b) shows that values of parameter maps in dark regions are higher than those in bright regions. Furthermore, in the red rectangular box, the headlights region of the cars is strongly exposed, while its body region is weakly exposed. This demonstrates that the pixelwise parameter maps can adaptively adjust their values based on different illumination conditions in an image.

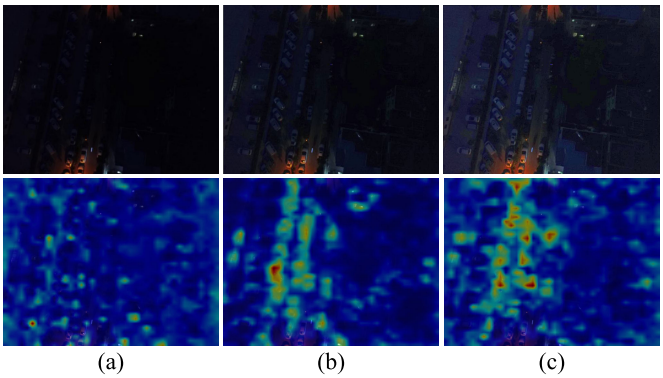


Fig. 12. Visualization of input images and their corresponding feature maps obtained from the backbone in detector. (a) Detector without performing exposure correction. (b) Detector "w/ Exposure x1" with the exposure module executed once. (c) Detector "w/ Exposure x4" with the exposure module executed four times.

4) Visualization of Feature Maps: We then visualize feature maps from the backbone of the Detection module to demonstrate the effectiveness of our progressive exposure strategy. To this end, we trained three detectors independently: a baseline detector referred to as "w/o Exposure," a detector with exposure executed once referred to as "w/ Exposure x1," and the detector with exposure executed four times referred to as "w/ Exposure x4." Fig. 12 illustrates the resulting feature maps obtained with the three detectors. Fig. 12(a) displays the input image and its feature maps obtained from the baseline detector without performing exposure correction. Fig. 12(b) and (c) presents the input images and their feature maps obtained executing the exposure correction module once and four times, respectively.

From this visualization, we can observe that the enhanced feature maps in Fig. 12(c) exhibit stronger activation values than these of the feature maps in Fig. 12(a) and (b). These visualization results demonstrate that our progressive exposure strategy enables the detector to focus more precisely on objects of interest in low-light images.

5) Effectiveness of Unsupervised Pretraining Strategy: We proceed to evaluate the unsupervised pretraining strategy adopted in our approach. We adopt our NightDrone dataset for unsupervised pretraining. We experimentally compare the detection performance of DEDet applying and without applying the proposed unsupervised pretraining strategy. The results displayed in the last two rows in Table II show that, when the FPP module is randomly initialized without pretraining, DEDet can achieve a good performance. With

TABLE III

PERFORMANCE COMPARISON ON THE NIGHTTIME DRONE-CAPTURED DATASETS, AND THE INPUT IMAGE SIZE IS 640 × 640 PIXELS. THE INFERENCE TIME OF MODELS IS MEASURED IN MILLISECONDS, AND ITS COMPUTATIONAL COST IS MEASURED IN GFLOPS

Method	GFLOPs	Time	DroneVehicle (Night)			VisDrone (Night)			NightDrone		
			AP[%]	AP_{50}[%]	AP_{75}[%]	AP[%]	AP_{50}[%]	AP_{75}[%]	AP[%]	AP_{50}[%]	AP_{75}[%]
Direct Training											
YOLOv5	108.1	12.9	29.3	55.5	27.8	25.2	41.2	25.3	30.8	58.3	29.1
TPH-YOLOv5	145.5	26.3	29.7	57.3	28.5	23.2	38.9	24.1	36.6	61.1	39.5
Separate Training											
SCI+YOLOv5	108.3	13.2	29.6	56.1	27.9	25.1	41.7	25.8	30.7	57.5	29.1
SCI+TPH-YOLOv5	145.7	26.6	29.5	56.9	28.3	23.0	39.0	23.9	36.7	61.3	39.6
Joint Training											
REGDet	438.3	79.9	32.7	58.8	32.9	24.0	39.1	25.2	38.3	61.2	40.2
IATDet	110.8	48.7	33.5	60.0	34.2	25.0	41.3	25.5	37.3	61.4	40.4
IA-YOLO	109.7	20.5	33.1	59.7	33.2	25.2	40.8	25.9	38.6	62.9	41.7
Our DEDet	129.8	36.7	**34.2**	**62.3**	**34.9**	**26.3**	**43.2**	**27.2**	**38.9**	**64.1**	**42.3**

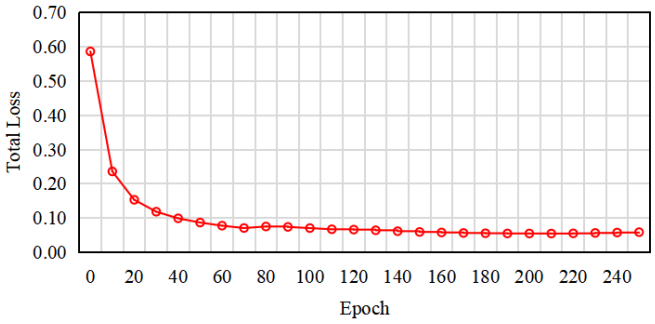


Fig. 13. Loss curve of the exposure module in the pretraining stage.

Fig. 14. Visualization results of the images exposed with the unsupervised pretrained strategy. Firstly, subfigures in the first row are original nighttime images. Secondly, subfigures in the second row are enhanced images. Note the obviously enhanced subregions highlighted in the red boxes.

our unsupervised pretraining strategy, DEDet achieved its best performance 64.1% of AP_{50} with a 1.4% performance gain.

Fig. 13 presents the loss curve of total loss $\mathcal{L}_t$ in (8) for pretraining the exposure module. It can be observed that the loss starts from a value of 0.59, gradually declines and finally stabilizes at a value of approximately 0.05. Furthermore, we also present the visual comparison between the input nighttime images and the images generated by the exposure module. Subfigures in the first row of Fig. 14 display the original nighttime images, and subfigures in the second row display images generated after applying the exposure module. These visualization results demonstrate that the exposure module effectively brightens the dark regions in the input images while preserving their image contents. Therefore, the above experimental results demonstrate that our fine exposure-correction module, with the unsupervised pretraining strategy, converges stably at the pretraining stage and effectively enhances low-light images.

D. Comparison With SOTA Methods

As introduced in Section II-C, the SOTA nighttime object detectors are grouped into three categories: the directly trained detectors, the separately trained detectors, and the jointly trained detectors. In this section, we conduct comparisons with SOTA methods in each category to demonstrate the strength of our approach. Tables III and IV present the comparison results

TABLE IV

PERFORMANCE COMPARISON ON THE EXDARK DATASET

Methods	AP[%]	AP_{50}[%]	AP_{75}[%]
YOLOv5	54.9	82.1	61.8
IATDet	54.8	82.5	62.1
IA-YOLO	55.4	83.3	62.8
Our DEDet	**56.2**	**84.8**	**63.2**

with the three groups of SOTA detectors on the four datasets. For a fair comparison, we replace the original detection modules of IATDet, REGDet, and IA-YOLO with the baseline detection model YOLOv5.

1) Comparison With Directly Trained Detectors: We first compare our DEDet with two directly trained detectors: YOLOv5 [8] and TPH-YOLOv5 [52]. Note that TPH-YOLOv5 is an object detector tailored for DroneDet.

Tables III and IV show that DEDet achieves the best detection performance on all four datasets. To be specific, for DroneVehicle (Night), DEDet brings 6.8% and 5.0% of AP_{50} higher than YOLOv5 and TPH-YOLOv5, respectively. For VisDrone (Night), DEDet brings 2.0% and 4.3% of AP_{50} higher than YOLOv5 and TPH-YOLOv5, respectively. For our NightDrone, DEDet brings 5.8% and 3.0% of AP_{50} higher

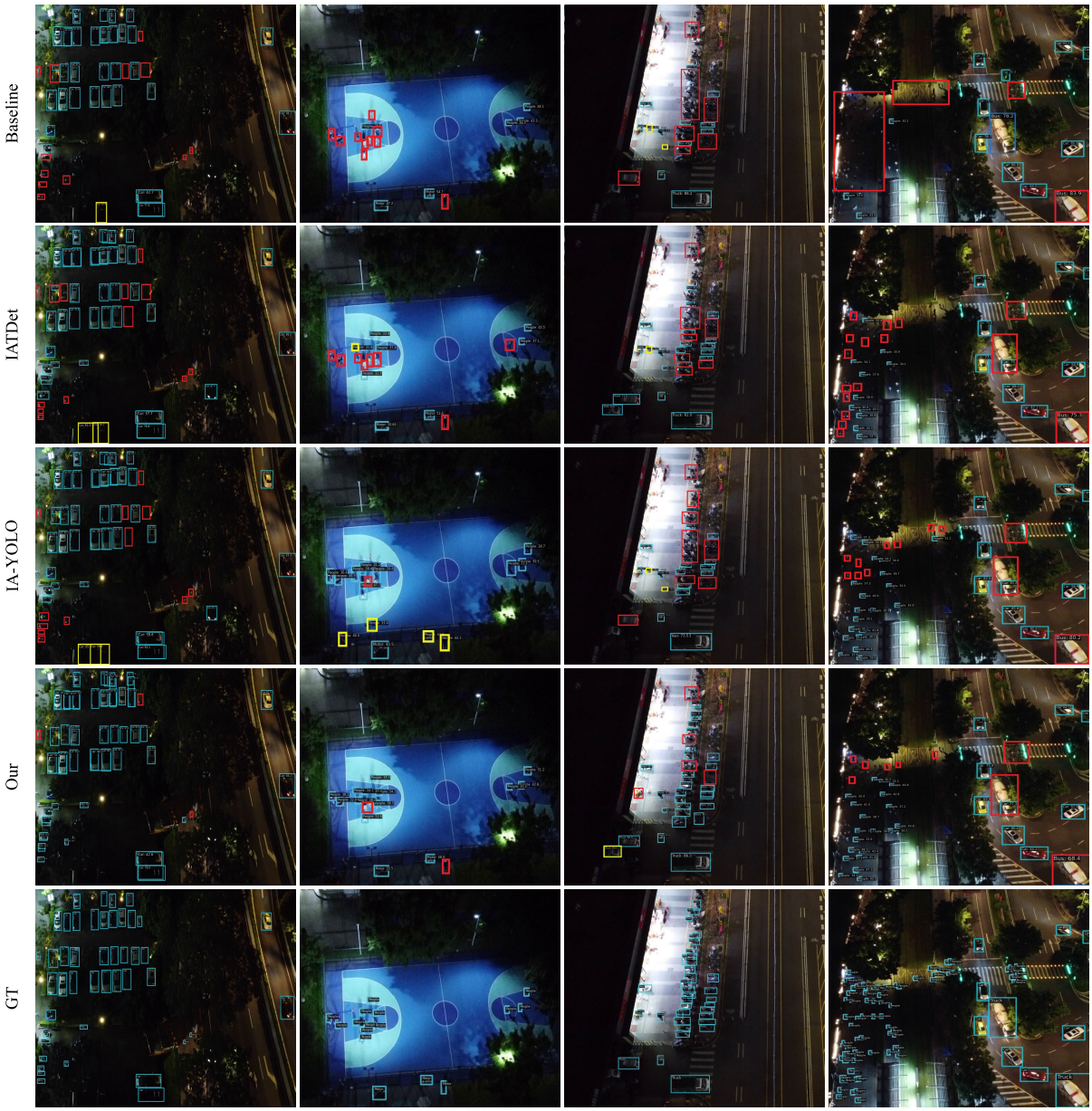


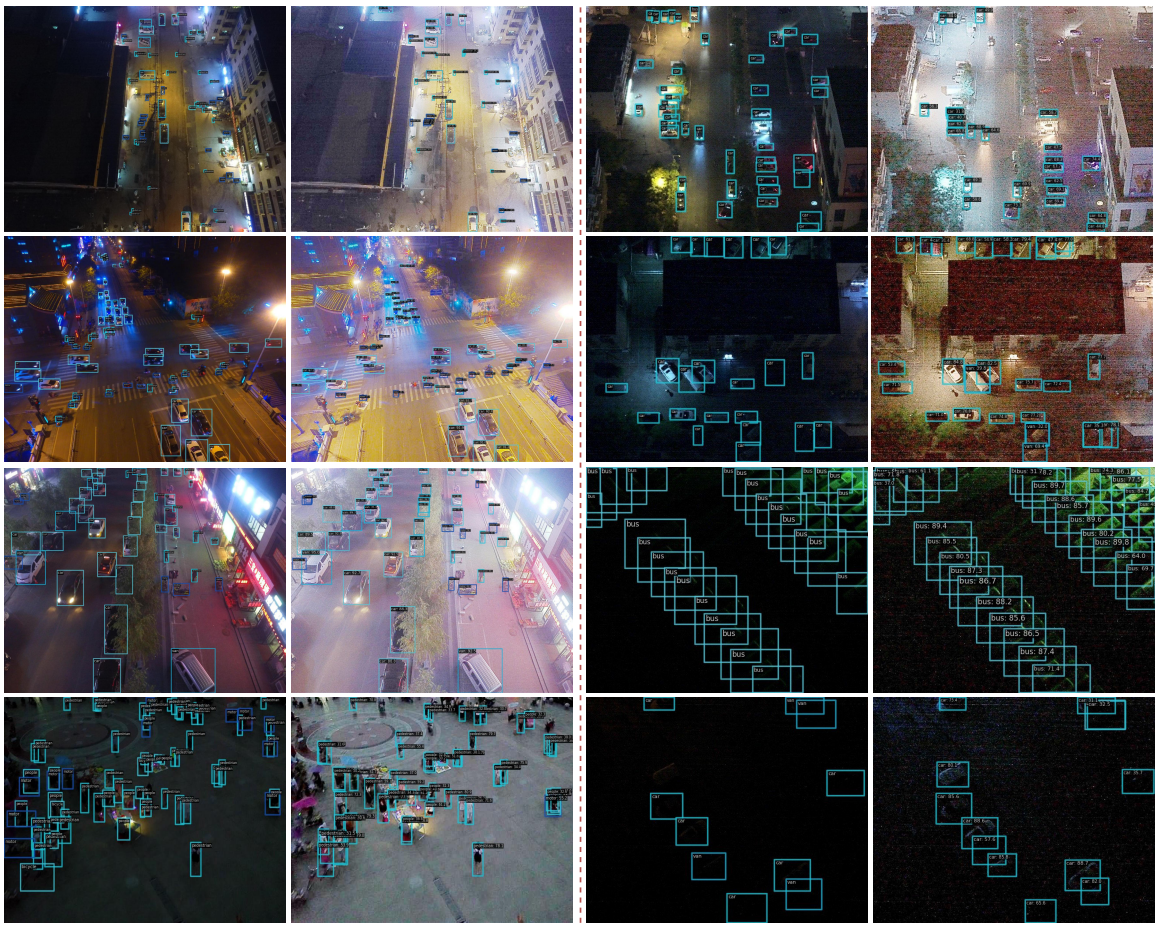
Fig. 15. Qualitative comparison of SOTA methods and our DEDet. The BBoxes, distinguished by different colors, represent various detection outputs: blue for true positives, red for missed targets, and yellow for false alarms.

than YOLOv5 and TPH-YOLOv5, respectively. For ExDark, DEDet brings 2.7% of AP_{50} higher than YOLOv5.

2) Comparison With Separately Trained Detectors: To compare with SOTA separately trained detectors, we adopted the exposure network SCI [50] to enhance the low-light images and then fed them into YOLOv5 [8] and TPH-YOLOv5 [52] for object detection, referred to as "SCI + YOLOv5" and "SCI + TPH-YOLOv5," respectively. Table III presents that DEDet also achieves the best detection performance on the three datasets. In particular, for DroneVehicle (Night), DEDet brings 6.2% and 5.4% higher than SCI + YOLOv5 and SCI + TPH-YOLOv5, respectively. For VisDrone (Night), DEDet brings 1.5% and 4.2% of AP_{50} gains to SCI + YOLOv5 and SCI + TPH-YOLOv5, respectively. For our NightDrone, DEDet brings 6.6% and 2.8% of AP_{50} gains to SCI + YOLOv5 and SCI + TPH-YOLOv5, respectively.

We observe that the separately trained paradigm does not necessarily improve DroneDet accuracy. Compared to the directly trained group, SCI + TPH-YOLOv5 even exhibits a slight degradation, while SCI + YOLOv5 shows some improvement. This suggests that the enhancement module may introduce random noise into features extracted by the detection module. Additionally, integrating Transformer and convolutional block attention module (CBAM) into the detector backbone of SCI + TPH-YOLOv5 increases the model's complexity, resulting in over-fitting to random noise.

3) Comparison With Jointly Trained Detectors: Finally, we compare our DEDet with three jointly trained models: IATDet [53], REGDet [11], and IA-YOLO [12]. IATDet utilizes attention queries to adjust image signal processor (ISP)-related parameters for subsequent detection. REGDet progressively generates images corresponding to various exposure settings to address nonuniform illumination and noise issues. IA-YOLO



(a) (b) (c) (d)

Fig. 16. Examples of the DEDet detection results from the real-world nighttime drone-captured datasets. (a) and (c) Original images with detection labels. (b) and (d) Corresponding exposed images generated by DEDet with the predicted BBoxes. Zoomed-in view on the small-size objects in low-light regions for a better viewing experience.

incorporates the differentiable image processing (DIP), which adopts the convolutional neural network-based parameter predictor (CNN-PP) to predict its filter parameters, to restore the latent information beneficial for the subsequent detection.

Tables III and IV demonstrate that DEDet achieves the best detection performance on all four datasets. To be specific, for DroneVehicle (Night), DEDet brings 2.3%, 3.5%, and 2.6% of AP_{50} higher than IATDet, REGDet, and IA-YOLO, respectively. For VisDrone (Night), the AP_{50} of DEDet is 1.9%, 4.1%, and 2.4% higher than those obtained by IATDet, REGDet, and IA-YOLO, respectively. For our NightDrone, the AP_{50} of DEDet is 2.7%, 2.9%, and 1.2% higher than those obtained by IATDet, REGDet, and IA-YOLO, respectively. For ExDark, the AP_{50} of DEDet is 2.3% and 1.5% higher than those obtained by IATDet and IA-YOLO, respectively.

We also observed that REGDet exhibits slow convergence and poor performance. Furthermore, during training the detection loss of the entire REGDet model struggles to escape the local optima. The other detector IA-YOLO with high accuracy merely adopts detection losses. Nonetheless, in the process of predicting hyperparameters of DIP by using CNN-PP, the datasets-specific upper and lower bounds of these hyperparameters need to be manually set carefully.

4) Visualization of Detection Results: Figs. 15 and 16 show the visual comparison of the detection results of DEDet and those obtained using SOTA methods. It can be observed that DEDet has achieved remarkable detection results on real-world nighttime drone-captured datasets. Thus, by outperforming SOTA methods across three different paradigms of solutions quantitatively and qualitatively, DEDet demonstrates its effectiveness and superiority in nighttime DroneDet tasks.

5) Computational Cost and Inference Time Analysis: To compare the computational complexity of DEDet with other SOTA methods, the floating point operations (GFLOPs) and inference time (milliseconds) are adopted and shown in Table III. As seen, DEDet has a processing time of 36.7 ms and GFLOPs of 129.8. Compared with the baseline model of YOLOv5, the models following the jointly trained strategy are more complex and have an increased inference time and GFLOPs due to the resources needed for LLIE. Moreover, our DEDet has obtained 2.6%, 2.4%, and 1.2% of AP_{50} performance gain on the DroneVehicle (Night), VisDrone (Night), and NightDrone datasets compared with IA-YOLO, which is a reasonable payback with its extra 18.3% computational cost.

VI. Conclusion

The proposed DEDet for nighttime DroneDet generates nighttime DroneDet-friendly images via detection-driven exposure correction. The proposed FPP module effectively addresses the issue of nonuniform illumination by estimating

pixelwise parameter maps of image filters. The proposed PFM module learns a highly sophisticated nonlinear mapping of pixel values from an original nighttime image to its DroneDet-friendly image by a progressively filtering strategy. Finally, we have created the NightDrone, the first dataset specifically designed for the task of nighttime DroneDet, which presents practical and challenging scenarios for detecting small objects under adverse illumination conditions. This dataset can extend the limits of DroneDet algorithms.

We can extend our DEDet as follows. First, we will incorporate semantic information into the enhancement module to estimate objectwise parameter maps for pixel value adjustment. This strategy has the potential to promote a positive correlation between LLIE and DroneDet performance. Second, we will collect and annotate more images in more diverse scenarios with drones to build the largest benchmark dataset for nighttime DroneDet. At last, considering that an infrared camera is robust to image degradation in extremely dark scenes, we intend to explore cross-modality detectors that leverage RGB-infrared images to improve the performance of nighttime DroneDet.

REFERENCES

[1] P. Ma, M. Macdonald, S. Rouse, and J. Ren, "Automatic geolocation and measuring of offshore energy infrastructure with multimodal satellite data," *IEEE J. Ocean. Eng.*, 2023, doi: 10.1109/JOE.2023.3319741.

[2] J. Ye, C. Fu, G. Zheng, D. P. Paudel, and G. Chen, "Unsupervised domain adaptation for nighttime aerial tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8886–8895.

[3] S. Zhang et al., "Person re-identification in aerial imagery," *IEEE Trans. Multimedia*, vol. 23, pp. 281–291, 2021.

[4] Y. Huang, J. Chen, and D. Huang, "UFPMP-Det: Toward accurate and efficient object detection on drone imagery," in *Proc. AAAI Conf. Artif. Intell.*, 2022, vol. 36, no. 1, pp. 1026–1033.

[5] V. Chalavadi, P. Jeripothula, R. Datla, S. B. Ch, and C. K. Mohan, "MSODANet: A network for multi-scale object detection in aerial images using hierarchical dilated convolutions," *Pattern Recognit.*, vol. 126, Jun. 2022, Art. no. 108548.

[6] S. Deng et al., "A global-local self-adaptive network for drone-view object detection," *IEEE Trans. Image Process.*, vol. 30, pp. 1556–1569, 2021.

[7] Y. Xi, W. Jia, Q. Miao, J. Feng, X. Liu, and F. Li, "CoDerainNet: Collaborative deraining network for drone-view object detection in rainy weather conditions," *Remote Sens.*, vol. 15, no. 6, p. 1487, Mar. 2023.

[8] G. Jocher. *YOLOv5 Source Code*. Accessed: Aug. 1, 2022. [Online]. Available: https://github.com/ultralytics/yolov5

[9] P. Zhu et al., "Detection and tracking meet drones challenge," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 7380–7399, Nov. 2022.

[10] T. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 740–755.

[11] J. Liang et al., "Recurrent exposure generation for low-light face detection," *IEEE Trans. Multimedia*, vol. 24, pp. 1609–1621, 2022.

[12] W. Liu, G. Ren, R. Yu, S. Guo, J. Zhu, and L. Zhang, "Image-adaptive YOLO for object detection in adverse weather conditions," in *Proc. AAAI Conf. Artif. Intell.*, 2022, vol. 36, no. 2, pp. 1792–1800.

[13] Y. Xi et al., "DRL-GAN: Dual-stream representation learning GAN for low-resolution image classification in UAV applications," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 1705–1716, 2021.

[14] Y. Bai, Y. Zhang, M. Ding, and B. Ghanem, "SOD-MTGAN: Small object detection via multi-task generative adversarial network," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 206–221.

[15] Y. Xi et al., "Beyond context: Exploring semantic similarity for small object detection in crowded scenes," *Pattern Recognit. Lett.*, vol. 137, pp. 53–60, Sep. 2020.

[16] H. Qiu et al., "Hierarchical context features embedding for object detection," *IEEE Trans. Multimedia*, vol. 22, no. 12, pp. 3039–3050, Dec. 2020.

[17] G. Li, Z. Liu, D. Zeng, W. Lin, and H. Ling, "Adjacent context coordination network for salient object detection in optical remote sensing images," *IEEE Trans. Cybern.*, vol. 53, no. 1, pp. 526–538, Jan. 2023.

[18] X. Dong, Y. Qin, Y. Gao, R. Fu, S. Liu, and Y. Ye, "Attention-based multi-level feature fusion for object detection in remote sensing images," *Remote Sens.*, vol. 14, no. 15, p. 3735, Aug. 2022.

[19] Y. Ye et al., "An adaptive attention fusion mechanism convolutional network for object detection in remote sensing images," *Remote Sens.*, vol. 14, no. 3, p. 516, Jan. 2022.

[20] Y. Xi, W. Jia, Q. Miao, X. Liu, X. Fan, and H. Li, "FiFoNet: Fine-grained target focusing network for object detection in UAV images," *Remote Sens.*, vol. 14, no. 16, p. 3919, Aug. 2022.

[21] J. Feng, K. Wang, Q. Miao, Y. Xi, and Z. Xia, "Personalized recommendation with hybrid feedback by refining implicit data," *Expert Syst. Appl.*, vol. 232, Dec. 2023, Art. no. 120855.

[22] Y. Sun, B. Cao, P. Zhu, and Q. Hu, "Drone-based RGB-infrared cross-modality vehicle detection via uncertainty-aware learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 10, pp. 6700–6713, Oct. 2022.

[23] Q. Wang, Y. Chi, T. Shen, J. Song, Z. Zhang, and Y. Zhu, "Improving RGB-infrared object detection by reducing cross-modality redundancy," *Remote Sens.*, vol. 14, no. 9, p. 2020, Apr. 2022.

[24] H. Farid, "Blind inverse gamma correction," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1428–1433, Oct. 2001.

[25] T. Arici, S. Dikbas, and Y. Altunbasak, "A histogram modification framework and its application for image contrast enhancement," *IEEE Trans. Image Process.*, vol. 18, no. 9, pp. 1921–1935, Sep. 2009.

[26] M. Li, J. Liu, W. Yang, X. Sun, and Z. Guo, "Structure-revealing low-light image enhancement via robust Retinex model," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2828–2841, Jun. 2018.

[27] R. Liu, L. Ma, J. Zhang, X. Fan, and Z. Luo, "Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10556–10565.

[28] C. Guo et al., "Zero-reference deep curve estimation for low-light image enhancement," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1777–1786.

[29] Y. Zhang, X. Guo, J. Ma, W. Liu, and J. Zhang, "Beyond brightening low-light images," *Int. J. Comput. Vis.*, vol. 129, no. 4, pp. 1013–1037, Apr. 2021.

[30] K. G. Lore, A. Akintayo, and S. Sarkar, "LLNet: A deep autoencoder approach to natural low-light image enhancement," *Pattern Recognit.*, vol. 61, pp. 650–662, Jan. 2017.

[31] Y. Zhang, J. Zhang, and X. Guo, "Kindling the darkness: A practical low-light image enhancer," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1632–1640.

[32] X. Xu, R. Wang, C.-W. Fu, and J. Jia, "SNR-aware low-light image enhancement," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17693–17703.

[33] W. Yang, W. Wang, H. Huang, S. Wang, and J. Liu, "Sparse gradient regularized deep Retinex network for robust low-light image enhancement," *IEEE Trans. Image Process.*, vol. 30, pp. 2072–2086, 2021.

[34] C. Li et al., "Detection-friendly dehazing: Object detection in real-world hazy scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 7, pp. 8284–8295, Jul. 2023.

[35] J. Yim and K.-A. Sohn, "Enhancing the performance of convolutional neural networks on quality degraded datasets," in *Proc. Int. Conf. Digit. Image Comput., Techn. Appl. (DICTA)*, Nov. 2017, pp. 1–8.

[36] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019, pp. 1–16.

[37] M. Singh, S. Nagpal, R. Singh, and M. Vatsa, "Dual directed capsule network for very low resolution image recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 340–349.

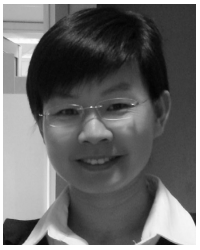
[38] Y. Li et al., "CBANet: An end-to-end cross-band 2-D attention network for hyperspectral change detection in remote sensing," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5513011.

[39] P. Ma et al., "Multiscale superpixelwise prophet model for noise-robust feature extraction in hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5508912.

[40] Y. Pei, Y. Huang, Q. Zou, Y. Lu, and S. Wang, "Does haze removal help CNN-based image classification?" in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 682–697.
[41] I. Morawski, "Enabling effective low-light perception using ubiquitous low-cost visible-light cameras," in *Proc. 30th ACM Int. Conf. Multimedia*, Oct. 2022, pp. 6915–6919.
[42] T. Ma, L. Ma, X. Fan, Z. Luo, and R. Liu, "PIA: Parallel architecture with illumination allocator for joint enhancement and detection in low-light," in *Proc. 30th ACM Int. Conf. Multimedia (ACM MM)*, Oct. 2022.
[43] X. Xue, J. He, L. Ma, Y. Wang, X. Fan, and R. Liu, "Best of both worlds: See and understand clearly in the dark," in *Proc. 30th ACM Int. Conf. Multimedia (ACM MM)*, Oct. 2022, pp. 2154–2162.
[44] T. Xiao, P. Dollar, M. Singh, E. Mintun, T. Darrell, and R. Girshick, "Early convolutions help transformers see better," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 30392–30400.
[45] R. Wang, Q. Zhang, C.-W. Fu, X. Shen, W.-S. Zheng, and J. Jia, "Underexposed photo enhancement using deep illumination estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6842–6850.
[46] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D, Nonlinear Phenomena*, vol. 60, nos. 1–4, pp. 259–268, Nov. 1992.
[47] T. Mertens, J. Kautz, and F. Van Reeth, "Exposure fusion: A simple and practical alternative to high dynamic range photography," *Comput. Graph. Forum*, vol. 28, no. 1, pp. 161–171, Mar. 2009.
[48] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A new backbone that can enhance learning capability of CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 1571–1580.
[49] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes challenge: A retrospective," *Int. J. Comput. Vis.*, vol. 111, no. 1, pp. 98–136, Jan. 2015.
[50] L. Ma, T. Ma, R. Liu, X. Fan, and Z. Luo, "Toward fast, flexible, and robust low-light image enhancement," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 5627–5636.
[51] Y. P. Loh and C. S. Chan, "Getting to know low-light images with the exclusively dark dataset," *Comput. Vis. Image Understand.*, vol. 178, pp. 30–42, Jan. 2019.
[52] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 2778–2788.
[53] Z. Cui et al., "You only need 90k parameters to adapt light: A light weight transformer for image enhancement and exposure correction," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2022, pp. 21–24.

Yue Xi received the Ph.D. degree from Northwestern Polytechnical University, Xi'an, China, in 2020, and the Ph.D. degree from the University of Technology Sydney, Sydney, NSW, Australia, in 2021.

He is currently a Lecturer with the Guangzhou Institute of Technology, Xidian University, Guangzhou, China. His research interests include image or video analysis, computer vision, and pattern recognition.



Wenjing Jia (Member, IEEE) received the Ph.D. degree in computing science from the University of Technology Sydney (UTS), Sydney, NSW, Australia, in 2007.

She is currently an Associate Professor with the Faculty of Engineering and Information Technology and a Core Research Member of the Global Big Data Technologies Centre, UTS. She has authored more than 200 quality journal articles and conference papers. Her research interests include image/video analysis, computer vision, and pattern recognition.

Qiguang Miao (Senior Member, IEEE) received the Ph.D. degree in computer application technology from Xidian University, Xi'an, China, in 2005.

He is currently a Professor and a Ph.D. Student Supervisor with the School of Computer Science and Technology, Xidian University. He has authored or coauthored more than 300 peer-reviewed journal articles or conferences papers. His research interests include machine learning, intelligent image processing, and malware behavior analysis and understanding.

Junmei Feng received the Ph.D. degree from Northwestern Polytechnical University, Xi'an, China, in 2020.

She is currently a Lecturer with the Guangzhou Institute of Technology, Xidian University, Guangzhou, China. Her research interests include information recommendation, virtual reality, and image analysis.

Jinchang Ren (Senior Member, IEEE) received the B.Eng., M.Eng., and D.Eng. degrees from Northwestern Polytechnical University, Xi'an, China, in 1992, 1997, and 2000, respectively, and the Ph.D. degree from the University of Bradford, Bradford, U.K., in 2019.

He is currently a Professor of computing with Robert Gordon University, Aberdeen, U.K. He has authored or coauthored more than 300 peer-reviewed journal articles or conference papers. His research interests include hyperspectral imaging, image processing, computer vision, big data analytics, and machine learning.

Heng Luo is currently pursuing the M.S. degree in computer science with the Guangzhou Institute of Technology, Xidian University, Guangzhou, China.

His research interests include machine learning, computer vision, and lightweight network.