Instance Segmentation in the Dark

Linwei Chen[1,2] · **Ying Fu**[2] · **Kaixuan Wei**[4] · **Dezhi Zheng**[1] · **Felix Heide**[3,5]

Abstract Existing instance segmentation techniques are primarily tailored for high-visibility inputs, but their performance significantly deteriorates in extremely low-light environments. In this work, we take a deep look at instance segmentation in the dark and introduce several techniques that substantially boost the low-light inference accuracy. The proposed method is motivated by the observation that noise in low-light images introduces high-frequency disturbances to the feature maps of neural networks, thereby significantly degrading performance. To suppress this "feature noise", we propose a novel learning method that relies on an adaptive weighted downsampling layer, a smooth-oriented convolutional block, and disturbance suppression learning. These components effectively reduce feature noise during downsampling and convolution operations, enabling the model to learn disturbance-invariant features. Furthermore, we discover that high-bit-depth RAW images can better preserve richer scene information in low-light conditions compared to typical camera sRGB outputs, thus supporting the use of RAW-input algorithms. Our analysis indicates that high bit-depth can be critical for low-light instance segmentation. To mitigate the scarcity of annotated RAW datasets, we leverage a low-light RAW synthetic pipeline to generate realistic low-light data. In addition, to facilitate further research in this direction, we capture a real-world low-light instance segmentation dataset comprising over two thousand paired low/normal-light images with instance-level pixel-wise annotations. Remarkably, without any image preprocessing, we achieve satisfactory performance on instance segmentation in very low light (4 % AP higher than state-of-the-art competitors), meanwhile opening new opportunities for future research. Our code and dataset are publicly available to the community[1].

Keywords Instance segmentation · feature denoising · low-light image dataset · object detection.

✉ Ying Fu
fuying@bit.edu.cn

1 Advanced Research Institute of Multidisciplinary Science, Beijing Institute of Technology, Beijing, China

2 School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China

3 Department of Computer Science, Princeton University, Princeton, New Jersey, USA

4 Department of Electrical and Computer Engineering, McGill University, Quebec, Canada

5 Algolux, Rue Richardson, Montreal, Canada

1 Introduction

Instance segmentation, as a technique that solves the problem of object detection and semantic segmentation at the instance level simultaneously, plays a critical role in helping computers understand visual information and thus supports applications such as robotics (Fang et al., 2018; Mohan and Valada, 2021) and autonomous driving (De Brabandere et al., 2017), and etc.

With the advent of deep learning, many learned instance segmentation methods have been proposed (Bolya et al., 2019; Chen et al., 2020, 2019b; He et al., 2017; Lee and Park, 2019), and have achieved promising performance in well-lit scenarios. However, these methods often fail to work well in dimly-lit environments, where the detailed contents are "buried" by severe noise caused by limited photon count and imperfections in photodetectors. While low-light instance

---

[1] https://github.com/Linwei-Chen/LIS

(a) Short-exposure camera output   (b) Result of enhanced image

(c) Our result from RAW data   (d) Long-exposure reference
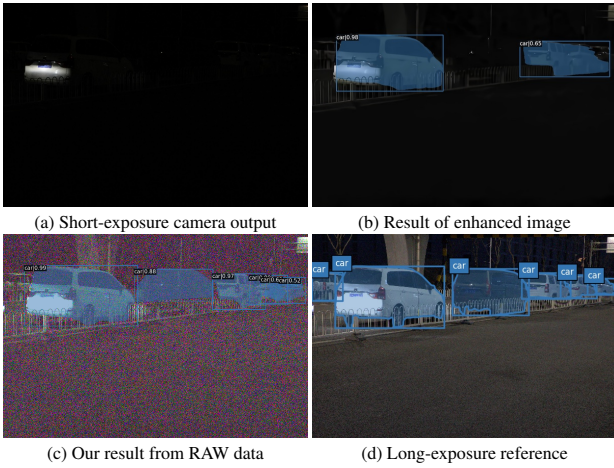
Fig. 1 Extreme low-light instance segmentation with Mask R-CNN (He et al., 2017) based upon (a) a short-exposure low-light image directly; (b) the image preprocessed by the state-of-the-art low-light enhancement algorithm (Guo et al., 2020) plus the denoising algorithm (Gu et al., 2019); (c) our proposed method on the amplified RAW image (the displayed result here is converted from RAW space to sRGB for visualization); and (d) the corresponding long-exposure reference image.

segmentation is an important task, there are few methods or datasets specifically designed for this purpose. Relevant low-light recognition/detection methods (Cui et al., 2021; Sasagawa and Nagahara, 2020) and datasets (Loh and Chan, 2019; Morawski et al., 2021; Yang et al., 2021) are still in their infancy. In this context, a common and simple solution is to combine image enhancement/denoising algorithms with instance segmentation models (Liu et al., 2020a). However, the additional image restoration process increases the computational cost and the overall latency of the pipeline. Even then, under extremely low light, these image restoration algorithms, as shown in Figure 1, can only recover limited scene information due to the permanent loss of image details in typical camera sRGB outputs.

In this work, we aim to craft a practical low-light instance segmentation framework in an end-to-end manner with marginal additional computational cost. To this end, we look deep into the instance segmentation model and analyze how low-light images harm the instance segmentation performance. We observe that the noise in low-light images brings "feature noise" (*i.e.*, high-frequency disturbance as shown in Figure 2(a)) into features inside the neural network. This leads to lower semantic responses of scene content in deep feature maps, therefore causing the low recall of scene content and degenerating the performance. This important phenomenon is also observed in adversarial defense/attack literatures (Szegedy et al., 2014; Xie et al., 2019), which suggests that restoring features from samples with adversarial noise can be critical for model robustness (Xie et al., 2019). Motivated by this observation, we propose to augment existing instance segmentation methods with an adaptive weight downsampling layer, smooth-oriented convolu-

tional block and disturbance suppression learning. They substantially improve the capability of models to learn noise-resisted features and thus boost the low-light segmentation accuracy appreciably. It is worth noting that they are model-agnostic and lightweight or even cost-free.

Specifically, the adaptive weight downsampling layer can generate content-aware low-pass filters during feature map downsampling. It aggregates local features adaptively and suppresses the high-frequency disturbance caused by noise as well as keeping the details in deep features. The smooth-oriented convolutional block enhances the ordinary convolutional layers by adding a smooth-oriented convolution branch. It helps to improve the robustness of the network for feature noise and can be re-parameterized (Ding et al., 2021) to the normal convolutional layer. The disturbance suppression learning guides networks to learn noise-resisted features, so as to keep stable semantic responses of scene content for noisy low-light images. Remarkably, they are model-agnostic. And only minor computational overhead is added by the adaptive weight downsampling layer, while smooth-oriented convolutional block and disturbance suppression learning introduce no extra computational cost since they are only involved during training.

Moreover, we notice that the high bit-depth can be crucial for low-light conditions. Thus to reduce the loss of scene information in dark conditions, instead of 8-bit sRGB camera outputs, we use 14-bit RAW sensor data as inputs, which have higher bit-depth and better potential to preserve scene information even under extreme low-light conditions (See Figure 2(b)). However, to date, there is no low-light RAW image dataset for instance segmentation, and its collection and annotation could be tremendously labor-intensive. To solve this, we leverage a low-light RAW synthetic pipeline. It can generate realistic RAW image datasets from any existing sRGB image datasets (*e.g.*, PASCAL VOC 2012 (Everingham et al., 2010), COCO (Lin et al., 2014a)), which makes an end-to-end training of RAW-input instance segmentation model feasible.

To systematically examine the performance of existing approaches under real low-light environments, we also capture and label a low-light instance segmentation (LIS) dataset with 2230 pairs of low/normal-light images, covering diverse real-world indoor/outdoor low-light scenes. Extensive experiments validate the superior instance segmentation performance of our method in the dark, consistently outperforming existing methods in accuracy and computation cost.

Our main contributions can be summarized as follows:

– We propose an adaptive weighted downsampling layer, smooth-oriented convolutional block and disturbance suppression learning to address the high-frequency disturbance within deep features that occurred in very low light. Interestingly, they also benefit the normal-lit instance segmentation.
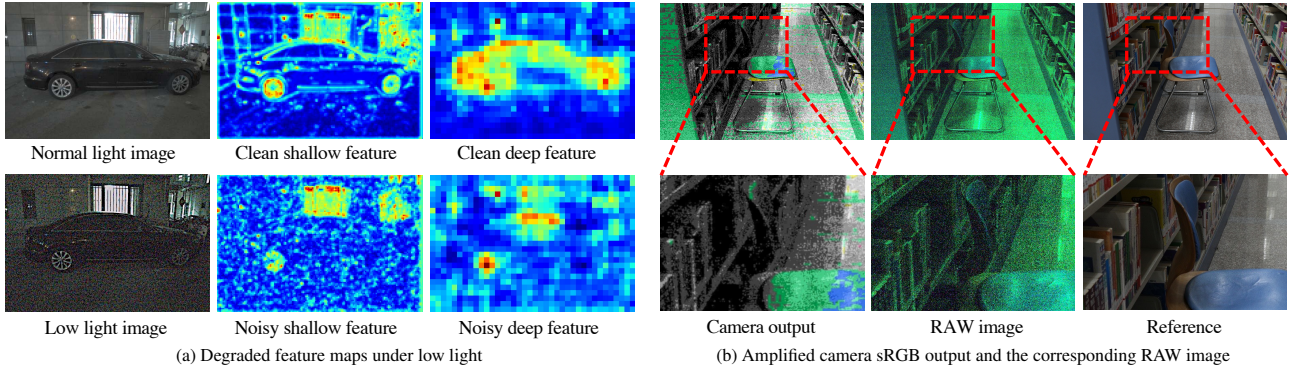
(a) Degraded feature maps under low light

(b) Amplified camera sRGB output and the corresponding RAW image

Fig. 2 Illustration of our key observations under dark regimes that drive our method design: (a) Degraded feature maps under low light. For clean normal-light images, the instance segmentation network is able to clearly capture the low-level (*e.g.*, edges) and high-level (*i.e.*, semantic responses) features of objects in shallow and deep layers, respectively. However, for noisy low-light images, shallow features can be corrupted and full of noise, and the deep features show lower semantic responses to objects. (b) Comparison between camera sRGB output and RAW image in the dark. Due to significantly low SNR, the 8-bit camera output loses much of the scene information, for example, the seat backrest structure is barely discernible, whereas is still recognizable in the RAW counterpart (**Zoom in for better details**).

– We exploit the potentials of RAW-input design for low-light instance segmentation and leverage a low-light RAW synthetic pipeline to generate realistic low-light RAW images from existing datasets, which facilitates end-to-end training.

– We collect a real-world low-light dataset with precise pixel-wise instance-level annotations, namely LIS, which covers more than two thousand scenes and can serve as a benchmark for instance segmentation in the dark. On LIS, our approach outperforms state-of-the-art competitors in terms of both segmentation accuracy and inference speed by a large margin.

2 Related Work

Normal instance segmentation. With the birth of deep learning, the field of computer vision has flourished (Fu et al., 2021; Wei et al., 2021; Zhang et al., 2022). Instance segmentation (Bolya et al., 2019; Chen et al., 2020, 2019b, 2021, 2022; He et al., 2017; Lee and Park, 2019) aim to predict the class label and the pixel-specific instance mask for objects. It localizes different classes of object instances present in various images. Many methods (Chen et al., 2019b; He et al., 2017) rely on Faster R-CNN (Ren et al., 2015) detector. By adopting the detect-then-refine strategy, they can achieve superior performance but run at a relatively slow speed. Other methods (Bolya et al., 2019; Chen et al., 2020; Lee and Park, 2019) are based on simple yet effective detectors (Lin et al., 2017b; Redmon et al., 2016; Tian et al., 2019), they can run in real-time and achieve competitive accuracy. Though achieving significant progress, most existing works only consider normal-light scenarios and largely overlooked low-light conditions.
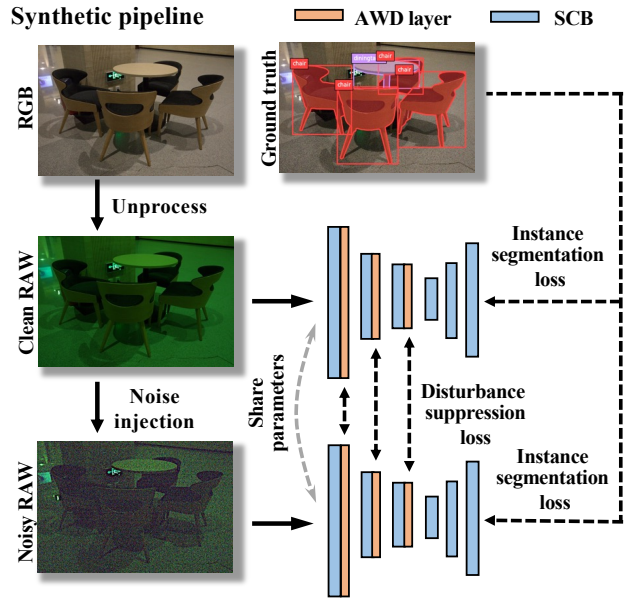

Fig. 3 Overview of our proposed method. The adaptive weighted downsampling (AWD) layer, smooth-oriented convolutional block (SCB), and disturbance suppression loss are designed to reduce the feature disturbance caused by noise, and the low-light RAW synthetic pipeline is employed to facilitate end-to-end training of instance segmentation on RAW images.

Instance segmentation in the dark. To adopt instance segmentation for very low-light, a straightforward solution is casting the low-light enhancement methods (Chen et al., 2018; Jiang et al., 2021; Lv et al., 2021; Yang et al., 2020a; Zhang et al., 2021b) or image denoising methods (Gu et al., 2019; Hahn et al., 2011; Hajiaboli, 2011; Liu et al., 2021b; Tan and Jiao, 2007; Ulyanov et al., 2020) as pre-processing steps. Compared with normal-light instance segmentation, research for low-light instance segmentation is at its early stage and relatively less at present.

A diverse body of work explores low-light classification. Gnanasambandam *et al.* (Gnanasambandam and Chan, 2020) present a new low-light image classification method using Quanta Image Sensors (QIS) and show promising results by utilizing a student-teacher learning scheme to classify the noisy QIS raw data.

As for low-light object detection, Liu *et al.* (Liu et al., 2020a) use a high-level vision model to guide the training of denoiser and demonstrate the benefit for image denoising and high-level vision tasks. Diamond *et al.* (Diamond et al., 2021) introduce Anscombe networks, which are lightweight neural camera ISP for demosaicking and denoising. It shows desirable performance on low-light classification by jointly learning Anscombe networks with classification networks. Julca-Aguilar *et al.* (Julca-Aguilar et al., 2021) propose a novel 3D object detection modality that exploits temporal illumination cues from a low-cost monocular gated imager. It shows better potential to deal with low-light or low-contrast regions. Wang *et al.* (Wang et al., 2021) propose a joint High-Low Adaptation (HLA) framework. By adopting a bidirectional low-level adaptation and multi-task high-level adaptation scheme, the proposed HLA-Face outperforms state-of-the-art methods even without using dark face labels for training. Sasagawa *et al.* (Sasagawa and Nagahara, 2020) propose glue layer to "glue" SID model (Chen et al., 2018) and YOLO model (Redmon et al., 2016) together. Cui *et al.* (Cui et al., 2021) propose to learn the intrinsic visual structure by encoding and decoding the realistic illumination-degrading transformation. They achieve desired performance on the low-light classification or low-light objection detection task but do not consider more challenging low-light instance segmentation.

Low-light synthesis. The low-light enhancement methods usually need low-light/normal-light image pairs for training (Chen et al., 2018; Lamba and Mitra, 2021; Lore et al., 2017; Wang et al., 2018a; Wei et al., 2018; Xiang et al., 2019; Zhang et al., 2021a), which is hard to obtain. Some works (Fu et al., 2022; Guo et al., 2020; Jiang et al., 2021) solve it by learning in a zero-reference way or utilizing unpaired images for training. And some works explore synthesizing low-light images from normal-light images. Retinex-Net (Wei et al., 2018) collects normal-light RAW images from RAISE (Dang-Nguyen et al., 2015) and makes their histogram of Y channel in YCbCr fit the result in low-light images from public datasets, thus getting synthetic low-light images with Adobe Lightroom. The GLADNet (Wang et al., 2018a) also synthesizes low-light images from RAW images in RAISE (Dang-Nguyen et al., 2015), which is done by adjusting the exposure, vibrance, and contrast parameters. And the recent works (Punnappurath et al., 2022; Xu et al., 2020) make progress in synthesizing low-light images by taking noise into consideration, but they still rely on RAW images that existing datasets for instance segmentation do not have.

Though some works (Cui et al., 2021; Lore et al., 2017) try to synthesize low-light sRGB images from normal-light ones, they only consider simple Gaussian and Poisson noise. Moreover, they are not applicable for synthesizing low-light RAW images from existing sRGB datasets. To solve this, we leverage unprocessing (Brooks et al., 2019) and employ a recently proposed physics-based noise model (Wei et al., 2020, 2021) to synthesize realistic low-light RAW images from any sRGB images with labels.

Datasets for low-light instance segmentation. Existing common datasets for instance segmentation, *e.g.*, PASCAL VOC (Everingham et al., 2010), cityscapes (Cordts et al., 2016) and COCO (Lin et al., 2014a), play an important role in the progress of instance segmentation algorithms under normal illumination. And there are several datasets available for nighttime detection (Liu et al., 2021a; Loh and Chan, 2019; Morawski et al., 2021; Yang et al., 2021) and semantic segmentation (Dai and Van Gool, 2018; Sakaridis et al., 2019; Tan et al., 2021). There are also some benchmark studies for understanding poor visibility environments (Dai et al., 2020; Fu et al., 2022; Sakaridis et al., 2018; Yang et al., 2020b). However, images in these datasets are captured in somewhat dim environments instead of extremely low-light, whose noise levels are low. Moreover, they are not suitable for instance segmentation due to the lack of instance-level pixel-wise labels. To better develop instance segmentation in extremely low-light, we collect and annotate a real-world low-light image dataset with precise pixel-wise instance-level annotations called Low-light Instance Segmentation (LIS).

3 Learning Segmentation in Low Light

The overview of the proposed method is shown in Figure 3. In this section, we first describe our motivation. Then we introduce the low-light RAW synthetic pipeline in Section 3.2. Finally, we show the details of the Adaptive Weighted Downsampling (AWD) layer, Smooth-oriented Convolutional Block (SCB), and Disturbance Suppression Learning (DSL) in Sections 3.3, 3.4 and 3.5, respectively.

3.1 Motivation

A practical low-light instance segmentation framework should be accurate and efficient. We notice the RAW images have better potential to recover scene information (see Figure 2 (b)), owing to higher bit depth. This should benefit the low-light instance segmentation. Nevertheless, collecting RAW image dataset for low-light instance segmentation is expensive and labor-intensive, and it is better if the existing normal-light image datasets (Everingham et al., 2010; Lin et al., 2014a) could be utilized for training target models. To this

end, we leverage unprocessing and noise injection to synthesize realistic low-light RAW images from any sRGB images with labels.

Furthermore, we observe noise in low-light images can disturb the prediction, and solutions of this degradation usually bring extra computational cost, *e.g.*, prepending enhancing/denoising step. To avoid this, instead of denoising the image, we aim to denoise the feature inside the instance segmentation model, *i.e.*, suppress the high-frequency disturbance within feature maps (see Figure 2 (a)). This should be more economical in computation than adding extra image-enhancing/denoising models. Next, we introduce the proposed method in detail.

3.2 Low-light RAW Synthetic Pipeline

Our low-light RAW synthetic pipeline consists of two steps, *i.e.*, unproccessing and noise injection. We introduce them one by one.

Unprocessing. Collecting a large-scale RAW image dataset is expensive and time-consuming, hence we consider utilizing existing sRGB image datasets (Everingham et al., 2010; Lin et al., 2014a). The sRGB image is obtained from RAW images by a series of image transformations of on-camera image signal processing (ISP), *e.g.*, tone mapping, gamma correction, color correction, white balance, and demosaicking. With the help of the unprocessing operation (Brooks et al., 2019), we can invert these image processing transformations, and RAW images can be obtained. In this way, we can create a RAW dataset with zero cost.

Noise injection. After obtaining clean RAW images by unprocessing, to simulate real noisy low-light images, we need to inject noise into RAW images. To yield more accurate results for real complex noise, we employ a recently proposed physics-based noise model (Wei et al., 2020, 2021), instead of the widely used Poissonian-Gaussian noise model (*i.e.*, heteroscedastic Gaussian model (Foi et al., 2008)). It can accurately characterize the real noise structures by taking into account many noise sources, including photon shot noise, read noise, banding pattern noise, and quantization noise.

3.3 Adaptive Weighted Downsampling Layer

To be robust to image noise, the features of networks should be clean and consistently respond to the scene content. As shown in Figure 2(a), noise in low-light images introduces high-frequency disturbance in feature maps of convolutional neural networks, which can mislead the following semantic information extraction and degrade the final prediction. We observe that the feature map downsampling is done by 1×1 convolution layers with a stride of 2 in wide-used vanilla

ResNet (He et al., 2016), as shown in Figure 4(b). This is similar to applying nearest neighbor interpolation for downsampling, which only considers the value of a single "pixel". It helps to reduce the computational cost but is useless for suppressing the noise in features. To better understand this, we show an example in Figure 5. We first obtain a noisy image from a clean one by injecting Gaussian noise ($\sigma = 60$), then downsample the noisy image by nearest neighbor interpolation and mean filter, respectively. Due to the local smoothness prior, the mean filter is able to suppress image noise during downsampling, whereas nearest neighbor interpolation can do nothing with noise. This applies to feature maps as well.

Downsample with low-pass filter. On the basis of this analysis, we propose to use a low-pass filter (*e.g.*, Gaussian filter, mean filter, or bilateral filter, with stride=2) for feature map downsampling. To verify if it is helpful for feature noise suppression, we evaluate the feature noise with

$$D(x, x', f(\cdot; \theta)) = \sum_{i=1}^{n} \|f^{(i)}(x; \theta) - f^{(i)}(x'; \theta)\|_2^2, \quad (1)$$

where D indicates feature disturbance caused by image noise, *i.e.*, feature noise, x and x' are clean normal-light image and corresponding noisy low-light image, and $f^{(i)}(x; \theta)$ is the i-th stage of feature maps in network $f(\cdot)$ with parameters of θ. As shown in Table 1, these low-pass filters are able to reduce the feature disturbance caused by noise in low-light images, and the instance segmentation performance also improves, which shows their effectiveness.

Learning to generate spatial-variant filter. Though these low-pass filters help to achieve better low-light instance segmentation results with minor extra computational cost, they are still suboptimal. For example, they may blur edge/texture features with relatively high frequency in the scene content. Furthermore, different spatial locations usually have different signal frequencies in feature maps. We need to apply different filters to them separately. Therefore, we propose to use spatial-variant filters, which can be formulated as

$$Y_{i,j} = \sum_{p,q \in S} W_{i,j}^{p,q} \cdot X_{i+p,j+q}, \quad (2)$$

where X, Y are input and output feature maps, (i, j) indicates the location in height, width dimensions, S points to the set of spatial locations surrounding (i, j), and W is the filter weight predicted by the network

$$
\begin{aligned}
V_{i,j} &= \phi(X_{\Psi_{i,j}}), \\
W_{i,j}^{p,q} &= \frac{\exp(V_{i,j}^{p,q})}{\sum_{p,q \in S} \exp(V_{i,j}^{p,q})},
\end{aligned}
\quad (3)
$$

where ϕ is the weight generation function, $\Psi_{i,j}$ indexes the set of pixels $V_{i,j}$ conditioned on. The softmax function can ensure the filter kernels are low-pass.
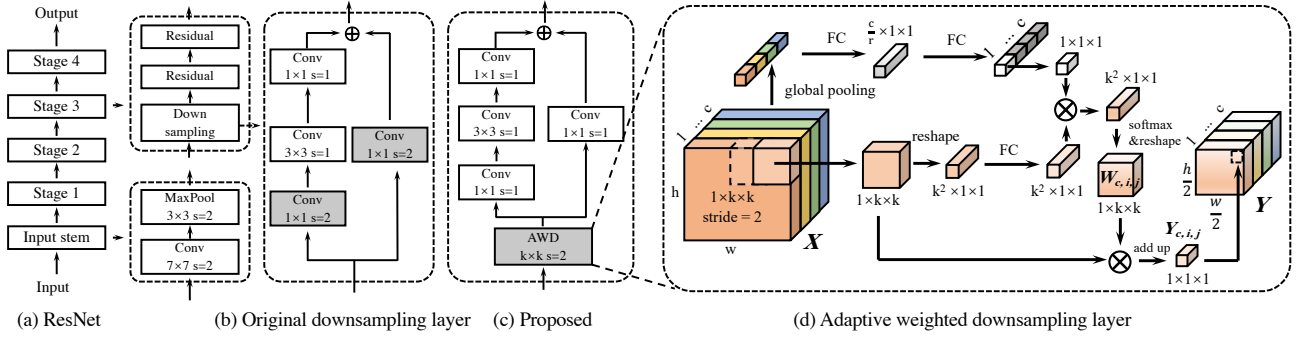
(a) ResNet   (b) Original downsampling layer   (c) Proposed   (d) Adaptive weighted downsampling layer

Fig. 4 Illustration of adaptive weighted downsample layer. The original downsampling operation in ResNet (He et al., 2016) is done by convolutional layers (b) with stride=2, which fails to leverage 3/4 of spatial features for feature noise suppression. The proposed adaptive weighted downsampling (AWD) layer (c) can selectively aggregate all surrounding features to generate downsampled features with less feature noise. FC in (d) indicates a fully connected layer, and "r" indicates the channel reduction ratio.

Table 1 Ablation study of different mechanisms for feature denoising during feature map downsampling.

| Filter type	Kernel size	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}	Disturbance	GFlops	Parameters
None	-	38.0	59.9	39.1	45.2	67.1	50.2	1.5292	109.95	43.78M
Gaussian	3×3	38.3	60.5	39.1	45.5	67.2	49.1	1.4264	109.96	43.78M
Bilateral	3×3	38.1	59.3	38.5	45.3	66.7	49.3	1.5288	109.96	43.78M
Mean	3×3	38.5	60.4	38.4	45.7	67.1	50.9	1.4524	109.96	43.78M
Spatial-variant	3×3	39.0	61.0	39.8	46.3	**68.0**	51.3	1.4011	110.08	43.93M
AWD	3×3	**39.3**	**61.4**	**40.2**	**46.4**	**68.0**	**51.6**	**1.3715**	110.25	44.65M

PSNR: 13.22   PSNR: 21.97   PSNR: 23.87
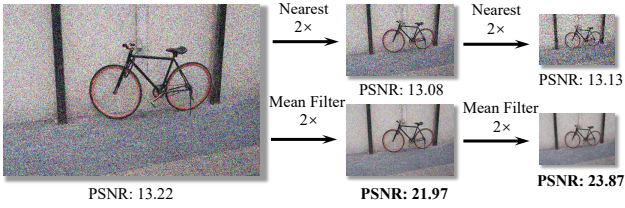
Fig. 5 Downsampling the noisy image with low-pass filters, *e.g.*, mean filter, can suppress the noise, whereas nearest neighbor interpolation cannot.
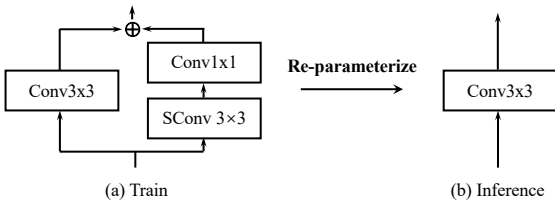

(a) Train   (b) Inference

Fig. 6 Illustration of the smooth-oriented convolutional block. SConv indicates the smooth-oriented convolution. (a) It explicitly employs a linear combination of multiple branches during the training stage. (b) And in the inference stage, it can be folded back to a normal 3×3 convolutional layer by using the re-parameterization technique (Ding et al., 2021).

Adaptive weighted downsampling layer. Table 1 shows the spatial-variant filter has superiority in feature denoising, but it still has two following drawbacks. First, considering different channels of the feature maps show the semantic responses to different image features, the signal frequencies can be variant across channels at the exact spatial locations. Thus spatial-variant is not enough to achieve

optimal results, it is necessary to generate spatial-variant and channel-variant filters for different spatial and channel locations. Second, the spatial-variant filter generates filter weights from local features and fails to utilize the context and global information. To solve these problems, we propose an adaptive weighted downsampling layer, which can be formulated as

$$Y_{c,i,j} = \sum_{p,q \in S} W^{p,q}_{c,i,j} \cdot X_{c,i+p,j+q}, \tag{4}$$

where (c, i, j) indicates the location in the channel, height, and width dimensions. The W is the filter weight predicted by the network

$$V_{c,i,j} = \phi(X_{\Psi_{c,i,j}}), \quad T_c = \phi'(\text{GP}(X)),$$
$$W^{p,q}_{c,i,j} = \frac{\exp(V^{p,q}_{c,i,j} \cdot T_c)}{\sum_{p,q \in S} \exp(V^{p,q}_{c,i,j} \cdot T_c)}, \tag{5}$$

where ϕ, ϕ' are also the weight generation functions, $\Psi_{c,i,j}$ indexes the set of pixels $V_{c,i,j}$ conditioned on, and GP is the global pooling operation. As illustrated in Figure 4(d), it is estimated by combining the local information V (by using local features) and global information T (by using global pooling features). The predicted T_c can adjust the smoothness of kernel $W_{c,i,j}$, which is similar to the temperature parameter in softmax (Hinton et al., 2015). To ensure that generated filters are low-pass, we use softmax to constrain weights to be positive and sum to 1. In this way, we predict content-aware low-pass filters for each position and channel, so as to keep the foreground signal and suppress the feature noise adaptively.

Fig. 7 Example scenes in our LIS dataset. Four image types (long-exposure normal-light and short-exposure low-light images in both RAW and sRGB formats) are captured for each scene.

3.4 Smooth-Oriented Convolutional Block

The adaptive weighted downsampling layer improves the robustness of networks with a carefully designed downsampling process. To further enhance the networks, we turn to focus on increasing the strength of ordinary convolutional blocks. The principles of convolutional block design are two sides. First, the improved convolutional block should be more robust to the feature noise. Second, the extra computational cost should be minor or free. To this end, we propose the smooth-oriented convolutional block. It can replace the ordinary 3×3 convolutional layers to enhance the network. Remarkably, it brings no extra computational cost during inference by using the re-parameterization technique (Ding et al., 2021).

The structure of the smooth-oriented convolutional block is shown in Figure 6, it explicitly employs a linear combination of multiple branches during the training stage. The main convolutional branch is the same as the original 3×3 convolutional layers. The auxiliary branch consists of a 1×1 convolutional layer and a smooth-oriented convolution, which can learn smooth kernels to suppress the feature noise in high-frequency. To ensure the filters are smooth, we use the mean filter or Gaussian filter to initialize the weights of smooth-oriented convolution and regularize its weights with the softmax function, so as to ensure that the learned 3×3 kernel for each channel is all positive and sum to 1. And it is followed by a 1×1 convolutional layer to fuse the filtered feature into the main convolutional branch.

And in the inference stage, the smooth-oriented convolutional block can be folded back to a normal 3×3 convolutional layer by the re-parameterization technique (Ding et al., 2021). Formally, we use $W \in R^{C_2 \times C_1 \times 3 \times 3}$ to de-

note the kernel of a 3×3 convolutional layer with C_1 input channels and C_2 output channels. The kernel of folded 3×3 convolutional can be re-parameterized as follows

$$W'_{3 \times 3}[i, j, h, t] = W_{3 \times 3}[i, j, h, t] \\ + (W_{1 \times 1}[i, j, 1, 1] * W_{\text{SConv}}[i, 1, h, t]), \quad (6)$$

where $i \in \{1, 2, ..., C_2\}$, $j \in \{1, 2, ..., C_1\}$, and $h, t \in \{1, 2, 3\}$ are indicators. $W'_{3 \times 3} \in R^{C_2 \times C_1 \times 3 \times 3}$ is the kernel weights of folded convolutional layer for inference, and $W_{3 \times 3} \in R^{C_2 \times C_1 \times 3 \times 3}$, $W_{1 \times 1} \in R^{C_2 \times C_1 \times 1 \times 1}$ and $W_{\text{SConv}} \in R^{C_2 \times 1 \times 3 \times 3}$ indicate 3×3 convolutional layer, 1×1 convolutional layer and smooth-oriented convolution during training, respectively.

3.5 Disturbance Suppression Learning

Ideally, a robust network should extract similar features regardless of whether the input image is corrupted by noise or not. Orthogonal to the architectural considerations, we introduce disturbance suppression learning to encourage the network to learn disturbance-invariant features during training. As shown in Figure 3, the total loss for learning is

$$L(\theta) = L_{\text{IS}}(x; \theta) + \alpha L_{\text{IS}}(x'; \theta) + \beta L_{\text{DS}}(x, x'; \theta), \quad (7)$$

where x is the unprocessed clean synthetical RAW image and x' is its noisy version, α and β are the weights of the losses. We empirically set α, β to 1, 0.01 for weighing. The L_{IS} is instance segmentation loss, which consists of classification loss, bounding box regression loss, and segmentation (per-pixel classification) loss. Its specific formula is related to the instance segmentation model, please refer to Mask R-CNN (He et al., 2017) for details. The model should learn to

Table 2 Ablation study for the low-light RAW synthetic pipeline on our LIS testing set. "UP" and "NI" indicate unprocessing and noise injection operations, respectively. The "*" indicates processing with gamma correction. All models are based on the vanilla Mask R-CNN (He et al., 2017) with the backbone of ResNet-50-FPN (He et al., 2016; Lin et al., 2017a).

Data Type	Training Set	Testing Set	UN	NI	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}
REAL	LIS sRGB-normal	LIS sRGB-normal	-	-	48.1	71.8	50.3	54.6	76.5	60.2
	LIS RAW-normal	LIS RAW-normal	-	-	45.4	70.0	46.6	52.8	74.7	63.4
	LIS RAW-normal*	LIS RAW-normal*	-	-	48.1	71.5	50.5	54.7	76.0	60.3
	LIS sRGB-dark	LIS sRGB-dark	-	-	35.5	57.5	36.1	42.9	64.3	46.1
	LIS RAW-dark	LIS RAW-dark	-	-	39.0	61.3	40.1	46.1	67.8	50.5
SYNTHETIC	COCO sRGB-normal	LIS RAW-dark	-	-	23.2	40.0	22.5	26.1	42.7	27.3
			√	-	23.4	38.0	23.3	26.6	42.5	28.7
			-	√	27.2	46.3	27.7	31.0	50.7	32.6
			√	√	**29.4**	**49.0**	**28.5**	**34.4**	**54.6**	**36.6**
	LIS sRGB-normal	LIS RAW-dark	-	-	31.6	50.2	31.6	36.6	56.0	39.3
			√	-	33.8	52.7	34.2	39.6	58.8	43.2
			-	√	35.1	55.5	35.4	40.6	61.1	44.1
			√	√	**38.0**	**59.9**	**39.1**	**45.2**	**67.1**	**50.2**

work stably whether the image is noisy or not. Hence L_{IS} is applied to both clean image x and noisy image x'. The L_{DS} is feature disturbance suppression loss, which is defined as

$$L_{DS}(x, x'; \theta) = \sum_{i=1}^{n} \|f^{(i)}(x; \theta) - f^{(i)}(x'; \theta)\|_2^2, \qquad (8)$$

where $f^{(i)}(x; \theta)$ is the i-th stage of feature maps of model. By minimizing the Euclidean distance between clean features $f^{(i)}(x; \theta)$ and noisy features $f^{(i)}(x'; \theta)$, disturbance suppression loss induces model to learn disturbance-invariant features. Therefore the feature disturbance caused by image noise can be reduced, and its robustness for corrupted low-light images is improved.

Different from perceptual loss (Gnanasambandam and Chan, 2020), we do not need to pretrain a teacher model, which makes our training simpler and faster. With $L_{IS}(x; \theta)$, $L_{IS}(x'; \theta)$, our model can learn discriminative features from both clean and noisy images, so as to keep stable accuracy no matter images are corrupted by noise or not. Whereas "student" in perceptual loss (Gnanasambandam and Chan, 2020) only sees noisy images, which leads to degradation on clean images and limits its robustness. Moreover, the domain gap of feature distribution between the teacher model and student model may harm the learning procedure. While we minimize the distance between clean features and noisy features predicted by the same model, which avoids this problem.

4 Low-light Instance Segmentation Dataset

Though evaluating on synthetic low-light images is a common and convenient practice (Cui et al., 2021), its results can severely deviate from the real world due to the much more complicated lighting conditions and image noise (Anaya and Barbu, 2018; Plotz and Roth, 2017). To reveal and systematically investigate the effectiveness of the proposed method in the real world, a real low-light image dataset for instance

segmentation is necessary and urgently needed. Considering there is no suitable dataset, therefore, we collect and annotate a Low-light Instance Segmentation (**LIS**) dataset using a Canon EOS 5D Mark IV camera. In Figure 7, we show some examples of annotated images in our LIS dataset. It exhibits the following characteristics:

- **Paired samples.** In the LIS dataset, we provide images in both sRGB-JPEG (typical camera output) and RAW formats, each format consists of paired short-exposure low-light and corresponding long-exposure normal-light images. We term these four types of images as *sRGB-dark, sRGB-normal, RAW-dark, and RAW-normal*. To ensure they are pixel-wise aligned, we mount the camera on a sturdy tripod and avoid vibrations by remote control via a mobile app.
- **Diverse scenes.** The LIS dataset consists of 2230 image pairs, which are collected in various scenes, including indoor and outdoor. To increase the diversity of low-light conditions, we use a series of ISO levels (*e.g.*, 800, 1600, 3200, 6400) to take long-exposure reference images, and we deliberately decrease the exposure time by a series of low-light factors (*e.g.*, 10, 20, 30, 40, 50, 100) to take short-exposure images for simulating very low-light conditions.
- **Instance-level pixel-wise labels.** For each pair of images, we provide precise instance-level pixel-wise labels annotated by professional annotators, yielding 10504 labeled instances of 8 most common object classes in our daily life (bicycle, car, motorcycle, bus, bottle, chair, dining table, tv).

We note that LIS contains images captured in different scenes (indoor and outdoor), and different illumination conditions. In Figure 7, object occlusion and densely distributed objects make LIS more challenging besides the low light.

Table 3 Ablation study for adaptive weighted downsampling (AWD), smooth-oriented convolutional block (SCB), and disturbance suppression learning (DSL). "Synthetic LIS" indicates using unprocessing and noise injection operations to synthetic low-light images from sRGB-normal images for training. "Real LIS" indicates the RAW-dark and RAW-normal image pairs in the training set are accessible for training. Results are reported on the LIS RAW-dark test set.

Training Data	Backbone	AWD	SCB	DSL	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}	GFlops	Parameters
	ResNet-50-FPN	-	-	-	38.0	59.9	39.1	45.2	67.1	50.2	109.95	43.78M
	ResNet-101-FPN	-	-	-	39.5	62.2	40.5	46.7	68.4	52.7	128.01	62.78M
Synthetic LIS	ResNet-50-FPN	√	-	-	39.3	61.4	40.2	46.4	68.0	51.6	110.25	44.65M
	ResNet-50-FPN	√	√	-	39.9	61.8	41.1	47.3	68.8	52.3	110.25	44.65M
	ResNet-50-FPN	√	√	√	**40.8**	**62.7**	**41.5**	**48.0**	**69.2**	**52.6**	110.25	44.65M
	ResNet-50-FPN	-	-	-	39.0	61.3	40.1	46.1	67.8	50.5	109.95	43.78M
	ResNet-101-FPN	-	-	-	40.7	63.6	41.4	48.7	70.2	53.5	128.01	62.78M
Real LIS	ResNet-50-FPN	√	-	-	41.0	63.6	41.5	48.6	70.8	51.9	110.25	44.65M
	ResNet-50-FPN	√	√	-	41.5	64.3	41.9	48.9	71.6	52.7	110.25	44.65M
	ResNet-50-FPN	√	√	√	**42.7**	**66.2**	**43.3**	**50.3**	**72.6**	**55.2**	110.25	44.65M

5 Experiments

In this section, we first introduce implementation details and evaluation metrics. Then we conduct ablation studies to evaluate the effectiveness of the proposed method. Finally, we compare our method against existing multi-step methods.

5.1 Implementation Details

All experiments here are conducted on Mask R-CNN (He et al., 2017) baseline with ResNet-50-FPN (He et al., 2016; Lin et al., 2017a) backbone for simplicity. Notice that our proposed method can be equipped with any network-based instance segmentation model.

Training details. Our framework is trained by synthetic low-light RAW-RGB[2] images generated from COCO (Lin et al., 2014b) dataset using our low-light RAW synthetic pipeline[3]. Our implementation is based on *MMDetection*. During training, we use random flip as data augmentation and train with a batch size of 8, a learning rate of 1e-2 for 12 epochs, with a learning rate dropping by $10\times$ at 8 and 11 epochs, respectively. To make the model quickly adapt to low-light settings, we use COCO pre-trained model as initialization.

Dataset and Evaluation Metrics. The real low-light instance segmentation performance is evaluated on the LIS dataset, in which the total 2230 image pairs are randomly split into a train set of 1561 pairs and a test set of 669 pairs.

Following (He et al., 2017), we measure the performance by using COCO-style AP (averaged over thresholds from 0.5 to 0.95 with an interval of 0.05), AP_{50} and AP_{75} (*i.e.*, AP at an IoU of 0.5). We also provide the results of detection,

Table 4 Ablation study for various bit-depth and encodings. The *sRGB-dark* with italics indicates images are obtained from corresponding RAW images. It is worth noting that we simulate various color encodings by quantizing the captured 14-bit RAW images to RAW images of different color encodings (*e.g.*, 8, 10, and 12 bits). This can be different from directly capturing images in corresponding bits.

Data type	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}
sRGB-dark (8-bit)	35.5	57.5	36.1	42.9	64.3	46.1
sRGB-dark (10-bit)	37.7	59.5	38.3	44.5	66.5	48.7
sRGB-dark (12-bit)	38.2	60.4	39.3	45.3	67.2	49.0
sRGB-dark (14-bit)	38.7	60.8	39.5	46.0	67.4	50.8
RAW-dark (8-bit)	35.5	58.2	35.8	42.7	65.6	45.2
RAW-dark (10-bit)	38.6	60.7	39.9	45.9	67.6	50.0
RAW-dark (12-bit)	39.0	61.4	40.0	46.0	68.5	50.3
RAW-dark (14-bit)	39.0	61.3	40.1	46.1	67.8	50.5

Table 5 Ablation for global pooling branch in adaptive weighted downsampling layer.

Method	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}
w/ Global Pooling	39.1	61.1	39.8	46.3	**68.0**	51.4
w/o Global Pooling	**39.3**	**61.4**	**40.2**	**46.4**	**68.0**	**51.6**

Table 6 Ablation study for kernel size of adaptive weighted downsampling layer. We use synthetic LIS training set for training. Results are reported on the LIS test set.

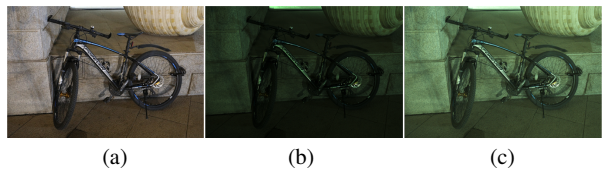| Kernel size	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}
None	38.0	59.9	39.1	45.2	67.1	50.2
2×2	38.9	61.3	39.1	**46.4**	**68.1**	51.0
3×3	**39.3**	61.4	**40.2**	**46.4**	68.0	**51.6**
4×4	38.9	**61.4**	39.5	**46.4**	**68.1**	50.2
5×5	38.4	60.5	39.4	45.5	67.6	50.6

(a)   (b)   (c)

Fig. 8 Illustration of sRGB-normal and RAW-normal with or without gamma correction. The sRGB-normal (a) is much visually brighter than the RAW-normal (b), especially for the dark region. And after gamma correction, RAW-dark (c) shows similar illumination to sRGB-normal.

which are represented as AP^{box}, AP^{box}_{50}, and AP^{box}_{75}. To evaluate inference speed, we measure Frame Per Second (FPS)

---

[2] To make the detector compatible with sRGB inputs, instead of the Bayer RAW images, we follow (Chen et al., 2019a) to use demosaicked 3-channel RAW-RGB images as inputs, where the green channel is obtained by averaging the two green pixels in each two-by-two Bayer block. In the following, we refer to "RAW" and "RAW-RGB" interchangeably.

[3] We use COCO samples belonging to the same 8 object classes in the LIS dataset.

Fig. 9 Visualization of filter weights predicted by adaptively weighted downsampling (AWD) layer. The top row shows input images, we convert RAW images to sRGB images for visualization. The bottom row shows the visualized standard variance of learned filter weights for each position, brighter color means a higher standard variance of predicted filter weights.

Table 7 Comparison and combination with different attention mechanisms. The baseline model is Mask R-CNN (He et al., 2017) with ResNet-50-FPN (He et al., 2016; Lin et al., 2017a). We compare and combine the proposed Adaptive Downsampling (AWD) layer with typical spatial attention (*i.e.*, non-local (Wang et al., 2018b)), channel attention (*i.e.*, squeeze-and-excitation (SE) (Hu et al., 2018)) and both spatial and channel attention (*i.e.*, CBAM (Woo et al., 2018)). Models are trained on synthetic LIS and evaluated on the LIS test set. $\Delta\uparrow$ indicates the extra improvement brought by the AWD.

Method	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}
baseline	38.0	59.9	39.1	45.2	67.1	50.2
+ AWD	39.3	61.4	40.2	46.4	68.0	51.6
$\Delta\uparrow$	**+1.3**	**+1.5**	**+1.1**	**+1.2**	**+0.9**	**+1.4**
Non-local	38.5	60.0	39.8	45.6	67.4	50.5
Non-local + AWD	39.5	61.4	40.2	46.6	68.8	51.7
$\Delta\uparrow$	**+1.0**	**+1.4**	**+0.4**	**+1.0**	**+1.4**	**+1.2**
SE	39.2	61.4	40.3	46.1	67.8	52.0
SE + AWD	40.3	62.8	41.4	47.2	69.4	52.5
$\Delta\uparrow$	**+1.1**	**+1.4**	**+1.1**	**+1.1**	**+1.6**	**+0.5**
CBAM	38.7	60.7	40.0	45.7	67.3	51.1
CBAM + AWD	40.0	62.8	40.5	47.4	69.6	52.8
$\Delta\uparrow$	**+1.3**	**+2.1**	**+0.5**	**+1.7**	**+2.3**	**+1.7**

Table 8 Ablation study for the smooth-oriented convolutional block. SConv indicates smooth-oriented convolution. We use synthetic LIS training set for training. Results are reported on the LIS test set.

Method	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}
Baseline (/w AWD)	39.3	61.4	40.2	46.4	68.0	51.6
Gaussian	39.4	61.6	39.8	46.5	68.2	51.4
Mean	39.4	61.7	40.2	46.6	68.6	51.4
SConv	**39.9**	**61.8**	**41.1**	**47.3**	**68.8**	**52.3**

Table 9 Ablation study for disturbance suppression learning (DSL) and perceptual loss (PL). We use synthetic LIS training set for training. Results are reported on the LIS test set.

Method	RAW-dark			RAW-normal		
	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
Baseline	39.9	61.8	41.1	44.7	68.2	45.7
w/ PL	40.2	61.9	40.7	45.1	68.4	45.9
w/ DSL	**40.8**	**62.7**	**41.5**	**46.6**	**69.5**	**48.6**

for each method on 600×400 images with a single RTX 3090.

Table 10 Ablation study for adaptive weighted downsampling (AWD), smooth-oriented convolution block (SCB), and disturbance suppression learning (DSL) under normal-light conditions. Models are trained on the COCO train set, and results are reported on the COCO val set.

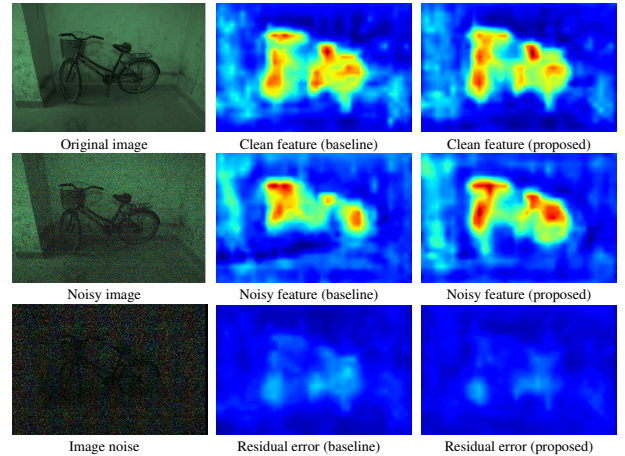| Method	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}
Mask R-CNN	34.4	55.6	36.9	38.0	58.6	41.5
+AWD	35.5	56.6	37.9	38.8	59.7	42.5
+AWD+SCB	35.5	56.6	38.0	39.0	59.9	42.5
+AWD+SCB+DSL	**36.1**	**57.4**	**39.0**	**39.5**	**60.3**	**43.2**

Fig. 10 Visualized high-level features. The residual error shows the difference between the clean feature and the noisy feature. It can be seen that the proposed method helps to reduce feature disturbance and keeps semantic responses to scene content when the image is noisy.

5.2 Ablation Studies

In this section, we first conduct ablation studies on input image types to reveal the advantage of RAW images for low-light instance segmentation and verify the effectiveness of low-light synthetic pipeline in Table 2. Then, we investigate the adaptive weighted downsampling layer, smooth-oriented convolutional block, and disturbance suppression learning. As shown in Table 3, they all contribute to performance improvement. Finally, we verify the effectiveness of the low-

light synthetic RAW pipeline. All results are reported on the LIS test set.

sRGB *vs.* RAW. To explore the upper bound of sRGB image and RAW image under normal-light and low-light conditions, we experiment with sRGB-normal, RAW-normal, sRGB-dark, and RAW-dark separately on our LIS dataset. As shown in Table 2, under normal light, sRGB and RAW have similar performance upper bound. But there is a consistent performance gap between RAW-normal and sRGB-normal (45.4 AP *vs.* 48.1 AP). To investigate deeper into this phenomenon, we notice that though they contain similar scene information, the sRGB-normal is much visually brighter than the RAW-normal, especially for the dark region (see Figure 8). This is caused by gamma correction in the pipeline of processing RAW to sRGB. Owing to the non-linear manner in perception, the sensitivity of humans to relative differences between darker tones is more significant than between lighter tones. And gamma correction can avoid allocating too many bits to highlights that humans cannot differentiate. Generally, the gamma correction can be written as:

$$I_{out} = I_{in}^{\gamma} \tag{9}$$

where $I \in [0, 1]$ is the normalized image pixel, and γ is usually set as $1/2.2$ for processing RAW to sRGB (Brooks et al., 2019). After applying gamma correction, RAW-dark with gamma correction shows similar illumination as shown in Figure 8 (c). And the corresponding performance is consistent with sRGB-normal, as shown in Table 2.

As for low-light conditions, the result of RAW images largely outperforms that of sRGB images, showing that RAW images keep richer scene information under very low light. This is critical for the instance segmentation task. We assume it is due to the higher color encodings of RAW images (14-bit RAW from Canon EOS 5D *vs.* 8-bit sRGB). To verify this, we simulate various color encodings by quantizing the captured 14-bit RAW images to RAW images of different color encodings (*e.g.*, 8, 10, and 12 bits). And the sRGB-dark of 10, 12, and 14-bit are obtained from corresponding RAW images with the image processing pipeline, which shows the necessity of RAW images. The image processing pipeline includes digital gain, white balance, demosaicing, color correction, and gamma correction. It is worth noting that quantizing the captured 14-bit RAW images to RAW images of various color encodings can be different from directly capturing them. However, since a specific commercial camera model only support RAW capture with constant bit depth (typically 14 bits in high-end DSLR), here, we simulate their results with quantization for fast verification.

As shown in Table 4, we can see the performance of both sRGB-dark and RAW-dark gradually increase from 8-bit to 14-bit, and the results of sRGB-dark are very similar to

RAW-dark. Besides, we also notice that the results of RAW-dark are slightly better than the sRGB-dark. The reason may be that the steps in the image processing pipeline can make the noise of sRGB more complex than RAW (Brooks et al., 2019), which leads to accuracy degradation. These quantitative results show that the high-bit property of RAW images plays a crucial part in low-light instance segmentation.

Low-light Synthetic RAW Pipeline. The unprocessing operation inverts sRGB images to synthetical RAW images, and noise injection simulates the corruption caused by limited photon count and imperfection of photodetectors. As shown in Table 2, they bring in 2.2 AP and 3.5 AP performance improvements, respectively. Moreover, when we combine these two steps together, the accuracy increases from 31.6 AP to 38.0 AP, which is very close to the result of training with real RAW-dark images 39.0 AP. It shows our synthetic pipeline is able to generate realistic RAW images. When we adopt the COCO dataset, the trend of results is similar and shows the satisfying generalization ability of low-light RAW synthetic pipeline.

Adaptive weighted downsampling (AWD) layer. Table 1 shows AWD reduces the feature disturbance compared with baseline, which means effectiveness on feature denoising. Though traditional filters can also be helpful, they may blur the foreground signal, which is not optimal. AWD avoids this problem by predicting content-aware filters. It surpasses all traditional low-pass filters and considerably improves the AP by 1.3 and 1.6 points when trained on LIS and COCO, respectively. And Table 5 verifies the effectiveness of the global pooling branch in the AWD.

To find out the best kernel size of the proposed AWD layer, we conduct experiments on different kernel sizes ranging from 2×2 to 5×5. As shown in Table 6, the AWD layer can bring 0.4-1.3 AP improvement, and the kernel size of 3×3 shows the best results, which is 1.3 AP better than the baseline.

Comparison with attention mechanisms. Here, we compare the AWD layer with attention mechanisms and discuss their essential differences. First, the AWD layer has a different motivation. Attention mechanisms are motivated by the human perception that treats information unequally. They assign different weights to input so as to pay more attention to important information. While the AWD layer is motivated by the fact that traditional low-pass filters (*e.g.*, Gaussian filter) can suppress high-frequency noise. To suppress high-frequency feature noise as well as keep the details, the AWD layer is designed to predict input-variant low-pass filters. Second, the AWD layer is technically different. Attention mechanisms such as (Woo et al., 2018) aim to improve the convolutional blocks while the AWD layer is proposed to improve the downsampling operation between convolutional blocks, which makes them orthogonal and complementary
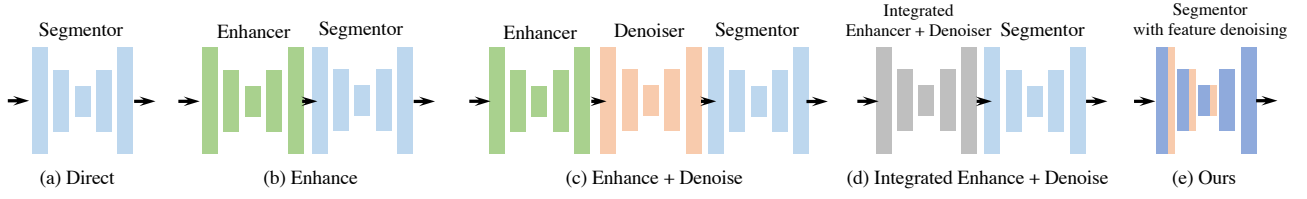
Fig. 11 Competing Methods. We illustrate different pipelines for comparison in Table 11. It can be seen that the proposed method is straight and concise.

Table 11 Quantitative comparisons of low-light instance segmentation. **To show the low-light performance in the uncontrolled real world, all models only use COCO data for training and evaluate on test set of the LIS dataset** Our method is trained on synthetic COCO (synthesized by the low-light RAW synthetic pipeline), and evaluated on RAW-dark images in the LIS dataset. Whereas the Mask R-CNN in other pipelines is optimized with original images in COCO, and evaluated on enhanced sRGB images outputted by the preprocessing methods in the LIS dataset.

Pipeline	Preprocessing method	Method	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}	FPS
Direct	-	Mask R-CNN	19.8	36.2	18.4	22.8	38.4	24.4	**56.2**
Enhance	HE (Gonzalez et al., 2002)	Mask R-CNN	18.9	33.0	18.4	22.5	36.5	24.1	42.7
	GLADNet (Wang et al., 2018a)	Mask R-CNN	14.0	24.4	13.8	16.0	25.9	17.5	31.2
	Retinex-Net (Wei et al., 2018)	Mask R-CNN	18.2	30.9	18.5	21.0	33.2	22.9	33.1
	EnlightenGAN (Jiang et al., 2021)	Mask R-CNN	19.0	33.3	18.6	22.3	36.5	24.2	38.0
	Zero-DCE (Guo et al., 2020)	Mask R-CNN	19.7	34.4	19.1	22.8	37.0	24.2	47.4
Enhance + Denoise	HE (Gonzalez et al., 2002) + SGN (Gu et al., 2019)	Mask R-CNN	21.0	36.4	20.7	25.1	40.2	26.5	31.4
	GLADNet (Wang et al., 2018a) + SGN (Gu et al., 2019)	Mask R-CNN	21.8	37.5	21.5	25.4	41.0	26.4	25.4
	Retinex-Net (Wei et al., 2018) + SGN (Gu et al., 2019)	Mask R-CNN	22.1	37.5	22.2	25.8	41.7	27.8	26.0
	EnlightenGAN (Jiang et al., 2021) + SGN (Gu et al., 2019)	Mask R-CNN	26.0	45.7	25.0	30.8	50.2	33.1	29.0
	Zero-DCE (Guo et al., 2020) + SGN (Gu et al., 2019)	Mask R-CNN	26.5	_46.1_	25.9	_31.2_	_50.4_	_33.9_	34.1
Integrated Enhance + Denoise	SID (Chen et al., 2018)	Mask R-CNN	_27.2_	45.4	_26.5_	30.9	49.6	32.9	43.8
	REDI (Lamba and Mitra, 2021)	Mask R-CNN	23.3	41.7	22.3	27.2	44.6	29.0	44.0
End-to-end (**Ours**)	-	Mask R-CNN	**31.8**	**52.3**	**31.4**	**37.6**	**58.4**	**40.4**	_53.1_

to each other. Therefore the AWD layer can steadily further improve the performance of the attention-based model as shown in Table 7. We compare the proposed AWD with CBAM (Woo et al., 2018) in Table 7. And we observe two interesting results that are consistent with the analysis. First, the AWD helps the baseline model achieve better results than CBAM (39.3 AP *vs.* 38.7 AP). Second, the AWD further improves the performance of CBAM by 1.3 (from 38.7 AP to 40.0 AP), which is the same degree of improvement as the baseline, *i.e.*, the AWD also improves the baseline by 1.3 (from 38.0 AP to 39.3 AP). These results verify that the improvement brought by AWD is entirely orthogonal to the CBAM. Furthermore, we conduct more experiments with different attention mechanisms including non-local (Wang et al., 2018b) and squeeze-and-excitation (SE) (Hu et al., 2018), which are spatial attention and channel attention, respectively. Their results in Table 7 draw the same conclusion, *i.e.*, the AWD helps the baseline model achieve better results than non-local and SE and the AWD can further steadily improve the results of non-local (Wang et al., 2018b) and SE (Hu et al., 2018).

Smooth-oriented convolutional block (SCB). The SCB explicitly employs a branch to learn to reduce the feature noise with a smooth filter. And at inference, the SCB can be folded to a normal convolutional layer, which means it boosts the model with no extra computational cost. Here, we try to replace the smooth-oriented convolution (SConv) with differ-

ent traditional smooth filters for comparison. As shown in Table 8, using Gaussian or mean filter also brings performance improvements for low-light instance segmentation. But due to their fixed filter weight, they fail to learn to deal with the feature noise in a flexible way. And SConv can be optimized during training, so as to learn the most appropriate filter weights for each channel and achieve better performance.

Disturbance suppression *vs.* Perceptual loss. Here, we compare our disturbance suppression learning with perceptual loss (Gatys et al., 2016; Gnanasambandam and Chan, 2020) for low-light instance segmentation. The perceptual loss (Gatys et al., 2016; Gnanasambandam and Chan, 2020) adopts a teacher-student structure for learning, its pretrained teacher extracts clean features from clean images to supervise the student for noisy images. Compared with perceptual loss (Gnanasambandam and Chan, 2020), the proposed disturbance suppression learning shows two beneficial characteristics. First, the disturbance suppression learning needs not to pretrain a teacher model, which makes our training simpler and faster. Second, the disturbance suppression learning can learn discriminative features from both clean and noisy images, whereas "student" in perceptual loss only sees noisy images and cannot fully utilize the clean images. Thus the disturbance suppression learning can keep stable accuracy no matter whether images are corrupted by noise or not. As shown in Table 9, on RAW-dark, the disturbance sup-

Table 12 Quantitative comparisons of low-light instance segmentation after finetuning on the LIS dataset. **All methods can access to train set of the LIS dataset for finetuning and are evaluated on test set of the LIS dataset.** Our method is trained on image pairs of RAW-dark and RAW-normal, and evaluated on RAW-dark images in the LIS dataset. While the Mask R-CNN in "Enhance + Denoise" and "Integrated Enhance + Denoise" pipelines use enhanced sRGB/RAW-dark images outputted by the preprocessing methods in the LIS dataset for training/finetuning and evaluation. The U-Net and Mask R-CNN in "Jointly optimized" pipeline are jointly optimized and evaluated on RAW-dark images in the LIS dataset, where the U-Net preprocessor is supervised by an extra image restoration ($L2$) loss using RAW-Normal images as ground truth.

Pipeline	Preprocessing method	Method	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}	FPS
Direct	-	Mask R-CNN	35.5	57.5	36.1	42.9	64.3	46.1	**56.2**
Direct (RAW)	-	Mask R-CNN	39.0	61.3	40.1	46.1	67.8	50.5	**56.2**
Enhance + Denoise	EnlightenGAN (Jiang et al., 2021) + SGN (Gu et al., 2019)	Mask R-CNN	37.1	60.2	37.4	44.5	67.0	48.6	29.0
	Zero-DCE (Guo et al., 2020) + SGN (Gu et al., 2019)	Mask R-CNN	36.9	60.3	37.4	44.8	67.5	49.0	34.1
Integrated	SID (Chen et al., 2018)	Mask R-CNN	37.8	60.0	38.3	44.7	66.6	46.9	43.8
Enhance + Denoise	REDI (Lamba and Mitra, 2021)	Mask R-CNN	36.0	59.0	35.8	42.8	66.1	45.9	44.0
Jointly optimized	U-Net (Ronneberger et al., 2015)	Mask R-CNN	<u>39.2</u>	<u>61.4</u>	<u>40.0</u>	<u>46.2</u>	<u>67.8</u>	<u>50.7</u>	43.8
End-to-end (**Ours**)	-	Mask R-CNN	**42.7**	**66.2**	**43.3**	**50.3**	**72.6**	**55.2**	<u>53.1</u>

pression learning increases AP by 0.9 point, while perceptual loss (Gnanasambandam and Chan, 2020) only brings 0.3 AP improvement. And on RAW-normal, the disturbance suppression learning shows 1.9 AP improvement while perceptual loss (Gnanasambandam and Chan, 2020) brings 0.4 AP. These results verify the above analysis.

Extra ablation studies on normal-light dataset. Moreover, we have conducted a series of experiments to evaluate the impact of AWD, SCB, and DSL on the normal-light dataset COCO (Lin et al., 2014a). We train the Mask R-CNN (He et al., 2017) on the COCO (Lin et al., 2014b) for 12 epochs, and the ResNet-50-FPN (He et al., 2016; Lin et al., 2017a) serves as backbone. As shown in 10, the AWD layer improves the performance by 1.0 AP without bells and whistles. It means the AWD layer also improves the robustness of networks under normal-light conditions by designing the downsampling process carefully. As for SCB, it adds a branch to learn smooth filters during training and can be folded back to a normal 3×3 convolutional layer by a linear combination. It helps the convolutional blocks to suppress the high-frequency feature noise caused by image noise. But the image noise is imperceptible in normal-light images of COCO (Lin et al., 2014a). Therefore, the SCB brings minor improvement on COCO (Lin et al., 2014b). We further evaluate the DSL for normal light images. We use the noise injection to synthetic low-light noise images. Interestingly, though it is proposed for the low-light task, it can also improve performance under normal light. The possible explanations are i) DSL pushes the model to learn noise-invariant features, which is more discriminative. ii) DSL makes model learn from both clean images and its noisy version, and the noisy images can be regarded as a kind of augmentation.

Visualization of learned filter weights. As shown in Figure 9, we visualize the learned filter weights predicted by the AWD layer. It can be seen that the predicted filter weights have a high standard variance for edges of the scene content and a low standard variance for the background. High variance corresponds to less blur, while low variance corresponds to more blur. This means the AWD layer can correctly predict content-aware filters to blur high-frequency content (e.g., edges of scene content) less to preserve foreground signals and blur low-frequency background more to suppress the feature noise.

Visualization of feature maps. As shown in Figure 10, we visualize the high-level features of networks. It can be seen that the proposed method helps to reduce feature disturbance and keep semantic responses to scene content when the image is noisy, which is important for precise low-light instance segmentation. This visualized result verifies the effectiveness of the proposed method.

Summary. To sum up, the RAW images show better potential than sRGB images for the low-light instance segmentation task. And the low-light synthetic RAW pipeline brings 6.2-6.4 AP improvements when only normal light sRGB images are available (see Table 2). Furthermore, the proposed method achieves 2.8-3.7 AP improvements to the vanilla model with minor extra computational cost, while replacing ResNet-50-FPN with ResNet-101-FPN only brings 1.5-1.7 AP improvements (see Table 3). These results substantially demonstrate both the effectiveness and efficiency of the proposed method.

5.3 Method Comparisons

In this section, we compare the proposed approach with three types of pipeline, *i.e.*, directly predicting on camera output, predicting on enhanced images, and predicting on enhanced and denoised images.

We select representative traditional (histogram equalization (Gonzalez et al., 2002)) and learning-based (GLAD-Net (Wang et al., 2018a), Retinex-Net (Wei et al., 2018), EnlightenGAN (Jiang et al., 2021), Zero-DCE (Guo et al., 2020), SID (Chen et al., 2018), REDI (Lamba and Mitra, 2021)) methods as enhancers and adopt the state-of-the-art SGN (Gu et al., 2019) as denoiser. Considering some competing methods (SID (Chen et al., 2018) and REDI (Lamba
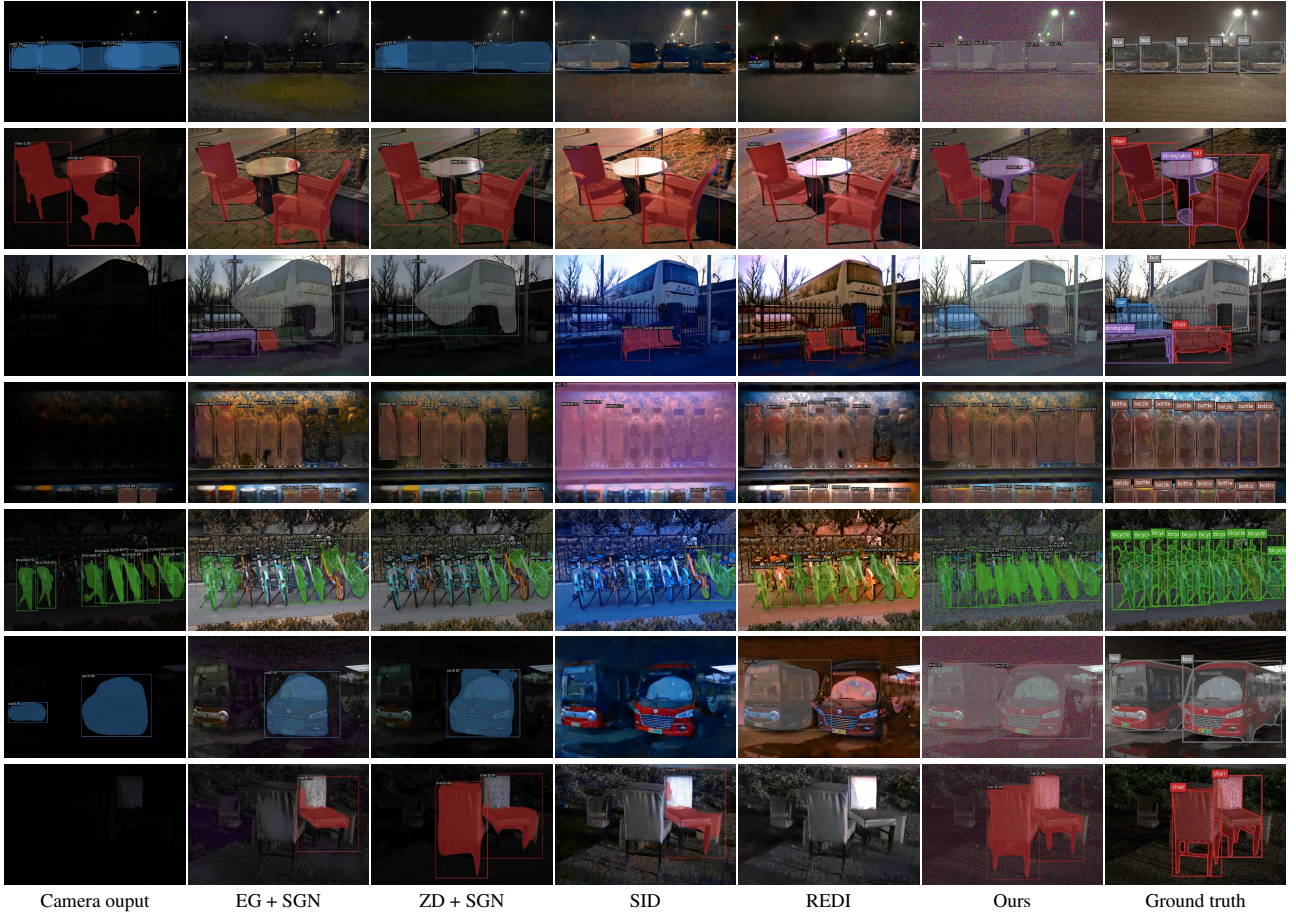
Fig. 12 Visual comparisons on our LIS dataset. To make it easy to browse, we only show visualized results of the baseline and four of the best competitors. Our RAW results are converted to sRGB for visualization.

Table 13 Quantitative comparisons of low-light instance segmentation when using Swin-T (Liu et al., 2021c) and ConvNeXt-T (Liu et al., 2022) as the backbone. And Mask R-CNN (He et al., 2017) serves as the instance segmentation model. For brevity, we choose four of the best competitors. Results are reported on the LIS test set. The setting for training and testing is the same as Table 11.

Pipeline	Preprocessing Method	Backbone	AP	AP_{50}	AP_{75}	AP^{box}	AP_{50}^{box}	AP_{75}^{box}
Direct	-	Swin-T	20.7	38.2	19.5	24.8	41.4	25.5
Enhance + Denoise	EnlightenGAN (Jiang et al., 2021) + SGN (Gu et al., 2019)	Swin-T	27.8	48.4	_28.3_	33.2	53.9	_36.3_
	Zero-DCE (Guo et al., 2020) + SGN (Gu et al., 2019)	Swin-T	_28.3_	_49.7_	27.8	_33.8_	_54.4_	_36.3_
Integrated	SID (Chen et al., 2018)	Swin-T	_28.3_	46.4	27.5	31.6	50.1	33.0
Enhance + Denoise	REDI (Lamba and Mitra, 2021)	Swin-T	25.7	43.7	25.3	29.8	48.3	31.2
End-to-end (**Ours**)	-	Swin-T	**32.6**	**53.5**	**32.3**	**37.8**	**59.2**	**40.9**
Direct	-	ConvNeXt-T	23.7	41.4	23.7	27.9	43.8	30.0
Enhance + Denoise	EnlightenGAN (Jiang et al., 2021) + SGN (Gu et al., 2019)	ConvNeXt-T	29.5	50.8	29.1	35.9	_56.2_	39.6
	Zero-DCE (Guo et al., 2020) + SGN (Gu et al., 2019)	ConvNeXt-T	30.2	_51.7_	30.2	36.1	_56.2_	_40.0_
Integrated	SID (Chen et al., 2018)	ConvNeXt-T	_31.8_	51.4	_31.7_	_36.6_	55.1	_40.0_
Enhance + Denoise	REDI (Lamba and Mitra, 2021)	ConvNeXt-T	27.6	46.6	27.5	32.2	49.9	35.1
End-to-end (**Ours**)	-	ConvNeXt-T	**36.8**	**58.5**	**36.9**	**42.7**	**64.0**	**47.4**

and Mitra, 2021)) already have explicit denoising mechanisms, so we do not append an extra denoising step. Notice that SID (Chen et al., 2018), REDI (Lamba and Mitra, 2021), and the proposed method are designed for taking RAW images as inputs while the rest of methods take sRGB images as inputs. All pipeline is illustrated in Figure 11.

For fairness, all settings use the same instance segmentation model (Mask R-CNN (He et al., 2017)). To accu-

rately reflect the practical use and unbiasedly evaluate the proposed method in real-world low-light environments, *we assume the LIS dataset is never seen by any methods during training*, *i.e.*, we regard the whole LIS dataset as a test set.

As shown in Table 11, without any preprocessing steps, the baseline normal instance segmentation model only has 19.8 AP, which shows limited accuracy in low-light conditions. After casting the enhancer to the pipeline, we intu-

Table 14 Quantitative comparisons of low-light instance segmentation when using PointRend (Kirillov et al., 2020) and Mask2Former (Cheng et al., 2022) as instance segmentation model. And ResNet-50-FPN (He et al., 2016; Lin et al., 2017a) serves as backbone. For brevity, we choose four of the best competitors. Results are reported on the LIS test set. The setting for training and testing is the same as Table 11.

Pipeline	Preprocessing method	Method	AP	AP_{50}	AP_{75}	AP^{box}	AP^{box}_{50}	AP^{box}_{75}
Direct	-	PointRend	20.6	37.2	19.2	23.5	39.3	25.1
Enhance + Denoise	EnlightenGAN (Jiang et al., 2021) + SGN (Gu et al., 2019)	PointRend	26.9	46.3	26.2	31.2	51.7	33.3
	Zero-DCE (Guo et al., 2020) + SGN (Gu et al., 2019)	PointRend	27.7	47.6	27.3	31.9	52.0	33.7
Integrated	SID (Chen et al., 2018)	PointRend	28.3	46.4	27.5	31.6	50.1	33.0
Enhance + Denoise	REDI (Lamba and Mitra, 2021)	PointRend	24.0	42.2	23.2	27.7	46.1	28.3
End-to-end (**Ours**)	-	PointRend	**32.8**	**52.9**	**39.8**	**37.1**	**57.9**	**39.8**
Direct	-	Mask2Former	21.4	37.9	20.9	22.9	36.9	23.2
Enhance + Denoise	EnlightenGAN (Jiang et al., 2021) + SGN (Gu et al., 2019)	Mask2Former	28.0	47.1	27.1	30.9	48.2	32.1
	Zero-DCE (Guo et al., 2020) + SGN (Gu et al., 2019)	Mask2Former	29.3	49.7	29.1	31.9	50.1	33.3
Integrated	SID (Chen et al., 2018)	Mask2Former	31.7	49.6	31.0	33.2	49.5	34.3
Enhance + Denoise	REDI (Lamba and Mitra, 2021)	Mask2Former	26.7	44.1	26.0	28.1	42.9	29.1
End-to-end (**Ours**)	-	Mask2Former	**35.6**	**55.2**	**35.2**	**37.8**	**55.9**	**39.9**

Table 15 Quantitative comparisons of low-light object detection. The backbone is ResNet-50-FPN (He et al., 2016; Lin et al., 2017a). Results are reported on the LIS test set. The setting for training and testing is the same as Table 11.

Pipeline	Preprocessing method	Method	AP^{box}	AP^{box}_{50}	AP^{box}_{75}
Direct	-	Faster R-CNN	21.9	37.4	22.4
Enhance	HE (Gonzalez et al., 2002)	Faster R-CNN	22.1	35.6	23.5
	GLADNet (Wang et al., 2018a)	Faster R-CNN	15.4	24.9	16.4
	Retinex-Net (Wei et al., 2018)	Faster R-CNN	19.6	31.1	21.5
	EnlightenGAN (Jiang et al., 2021)	Faster R-CNN	21.1	34.8	21.9
	Zero-DCE (Guo et al., 2020)	Faster R-CNN	22.0	35.9	23.5
Enhance + Denoise	HE (Gonzalez et al., 2002) + SGN (Gu et al., 2019)	Faster R-CNN	25.1	39.8	26.9
	GLADNet (Wang et al., 2018a) + SGN (Gu et al., 2019)	Faster R-CNN	24.1	39.1	25.3
	Retinex-Net (Wei et al., 2018) + SGN (Gu et al., 2019)	Faster R-CNN	25.5	41.0	27.4
	EnlightenGAN (Jiang et al., 2021) + SGN (Gu et al., 2019)	Faster R-CNN	29.5	48.5	30.4
	Zero-DCE (Guo et al., 2020) + SGN (Gu et al., 2019)	Faster R-CNN	30.5	49.8	32.5
Integrated	SID (Chen et al., 2018)	Faster R-CNN	30.1	47.6	32.3
Enhance + Denoise	REDI (Lamba and Mitra, 2021)	Faster R-CNN	29.8	47.9	31.6
End-to-end (**Ours**)	-	Faster R-CNN	**36.3**	**56.6**	**39.4**

itively expect performance improvement, but the accuracies stay the same (with histogram equalization (Gonzalez et al., 2002) and Zero-DCE (Guo et al., 2020)) or even decrease (with GLADNet (Wang et al., 2018a), Retinex-Net (Wei et al., 2018) and EnlightenGAN (Jiang et al., 2021)). We guess the reason is that these enhancers only improve the overall brightness but cannot handle the noise. To verify it, we further introduce denoiser to the pipeline, and the overall accuracy significantly increases as expected, *e.g.*, Zero-DCE (Guo et al., 2020) plus SGN (Gu et al., 2019) leads to 6.7 AP gain. Notice that these methods for comparison use camera outputs. Then, we also perform experiments with SID (Chen et al., 2018) and REDI (Lamba and Mitra, 2021), which can restore sRGB images from low-light RAW images. And numerical results are surprisingly good, *i.e.*, 27.2 AP with SID (Chen et al., 2018), which outperforms baseline by 7.4 points. This implies the superiority of using RAW images.

Though these enhancing and denoising steps boost the low-light instance segmentation performance remarkably, our method achieves the best quantitative results without extra preprocessing steps. Besides, the inference speed of the pro-

posed method outperforms all other pipelines. And its speed is very close to the original Mask R-CNN (He et al., 2017). Moreover, qualitative results illustrated in Figure 12 show the proposed method can consistently recall most of the targets even in challenging scenarios.

5.4 Finetuning on LIS Dataset

To compare the proposed approach more comprehensively, we choose four of the best pipelines in Table 11 for further comparison. Here, all methods can access to train set of the LIS dataset for finetuning, and are evaluated on test set of the LIS dataset. Moreover, we implement a jointly optimized pipeline (U-Net + Mask R-CNN, the U-Net (Ronneberger et al., 2015) is trained to recover clean normal light images from low-light images) for competition.

The results are shown in Table 12. It can be seen that all pipelines show better performance with the help of real low-light image pairs in the LIS dataset. This shows the low-light image dataset is important and necessary for solving the low-light instance segmentation task. The jointly opti-

mized pipeline outperforms all other pipelines but still underperforms compared with the proposed method by a large margin. Moreover, our solution shows a much higher inference speed than the jointly optimized and other pipelines.

5.5 Evaluation with Different Instance Segmentation Model

We note that the proposed method is model-agnostic, *i.e.*, it should work well with existing methods (Bolya et al., 2019; Chen et al., 2020, 2019b; Cheng et al., 2022; He et al., 2017; Huang et al., 2019; Kirillov et al., 2020; Lee and Park, 2019). Here, in addition to Mask R-CNN (He et al., 2017), we perform a series of extra experiments with CNN-based PointRend (Kirillov et al., 2020) and recent transformer-based Mask2Former (Cheng et al., 2022).

Similarly, as shown in Table 14, casting enhancers (EnlightenGAN (Jiang et al., 2021) and Zero-DCE (Guo et al., 2020)) and denoiser (Gu et al., 2019) to the pipeline can bring significant improvement. We also compare our method with SID (Chen et al., 2018) and REDI (Lamba and Mitra, 2021), which can restore sRGB images from RAW images. Though they largely boost performance compared with the baseline, our method achieves the best results. Notice that our method does not see any real raw images during training and can infer without any preprocessing steps. But SID (Chen et al., 2018) and REDI (Lamba and Mitra, 2021) require extra real-world paired RAW images for training.

5.6 Evaluation with Different Backbones

To further verify the proposed method, we also conduct experiments with recent great works of backbones, including the transformer-based Swin Transformer (Liu et al., 2021c) and CNN-based ConvNeXt (Liu et al., 2022).

As shown in Table 13, though the proposed smooth-oriented convolutional block is not available for the transformer-based Swin Transformer (Liu et al., 2021c), the proposed method still shows consistent advantages compared with other competitors. And the results with ConvNeXt (Liu et al., 2022) also draw the same conclusion.

5.7 Extension to Low-Light Object Detection

In addition to low-light instance segmentation, the proposed method and dataset can also apply to object detection (Liu et al., 2020b) in the low-light environment. We perform our experiments with the classic Faster R-CNN (Ren et al., 2015) detector, and results are provided in Table 15. We can see that the trend and conclusion are similar to that of low-light instance segmentation. It shows the effectiveness and generalization of the proposed method.

6 Conclusion

This paper explores end-to-end instance segmentation in very low light on RAW images. To deal with model degradation in low-light images, we propose adaptive weight downsample layer, smooth-oriented convolutional block, and disturbance suppression learning to handle the feature noise caused by notorious noise in low-light images. They can reduce feature noise during downsampling and convolution operation, and help the model learn disturbance-invariant features, respectively. Noticeably, the proposed method outperforms state-of-the-art competitors by a large margin with less computational cost.

Moreover, we also collect and annotate a large-scale real-world low-light instance segmentation dataset, which contains more than two thousand paired low/normal-light images with instance-level pixel-wise annotations. It can serve as a benchmark for high-level tasks in low-light conditions. We hope that our dataset and the experimental findings can inspire more work on vision in extremely low light in future research.

References

Anaya J, Barbu A (2018) Renoir–a dataset for real low-light image noise reduction. Journal of Visual Communication and Image Representation 51(1):144–154

Bolya D, Zhou C, Xiao F, Lee YJ (2019) Yolact: real-time instance segmentation. In: Proceedings of IEEE International Conference on Computer Vision, pp 9157–9166

Brooks T, Mildenhall B, Xue T, Chen J, Sharlet D, Barron JT (2019) Unprocessing images for learned raw denoising. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 11036–11045

Chen C, Chen Q, Xu J, Koltun V (2018) Learning to see in the dark. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 3291–3300

Chen C, Chen Q, Do MN, Koltun V (2019a) Seeing motion in the dark. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 3185–3194

Chen H, Sun K, Tian Z, Shen C, Huang Y, Yan Y (2020) Blendmask: Top-down meets bottom-up for instance segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 8573–8581

Chen K, Pang J, Wang J, Xiong Y, Li X, Sun S, Feng W, Liu Z, Shi J, Ouyang W, et al. (2019b) Hybrid task cascade for instance segmentation. In: Proceedings of IEEE International Conference on Computer Vision, pp 4974–4983

Chen L, Fu Y, You S, Liu H (2021) Efficient hybrid supervision for instance segmentation in aerial images. Remote Sensing 13(2):252

Chen L, Fu Y, You S, Liu H (2022) Hybrid supervised instance segmentation by learning label noise suppression. Neurocomputing 496:131–146

Cheng B, Misra I, Schwing AG, Kirillov A, Girdhar R (2022) Masked-attention mask transformer for universal image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 1290–1299

Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B (2016) The cityscapes dataset for semantic urban scene understanding. In: Proceedings of IEEE International Conference on Computer Vision, pp 3213–3223

Cui Z, Qi GJ, Gu L, You S, Zhang Z, Harada T (2021) Multitask aet with orthogonal tangent regularity for dark object detection. In: Proceedings of IEEE International Conference on Computer Vision, pp 2553–2562

Dai D, Van Gool L (2018) Dark model adaptation: Semantic image segmentation from daytime to nighttime. In: Proceedings of International Conference on Intelligent Transportation Systems, pp 3819–3824

Dai D, Sakaridis C, Hecker S, Van Gool L (2020) Curriculum model adaptation with synthetic and real data for semantic foggy scene understanding. International Journal of Computer Vision 128(5):1182–1204

Dang-Nguyen DT, Pasquini C, Conotter V, Boato G (2015) Raise: A raw images dataset for digital image forensics. In: Proceedings of the 6th ACM multimedia systems conference, pp 219–224

De Brabandere B, Neven D, Van Gool L (2017) Semantic instance segmentation for autonomous driving. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp 7–9

Diamond S, Sitzmann V, Julca-Aguilar F, Boyd S, Wetzstein G, Heide F (2021) Dirty pixels: Towards end-to-end image processing and perception. ACM Transactions on Graphics 40(3):1–15

Ding X, Zhang X, Ma N, Han J, Ding G, Sun J (2021) Repvgg: Making vgg-style convnets great again. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 13733–13742

Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A (2010) The pascal visual object classes (voc) challenge. International Journal of Computer Vision 88(2):303–338

Fang K, Bai Y, Hinterstoisser S, Savarese S, Kalakrishnan M (2018) Multi-task domain adaptation for deep learning of instance grasping from simulation. In: Proceedings of IEEE International Conference on Robotics and Automation, pp 3516–3523

Foi A, Trimeche M, Katkovnik V, Egiazarian K (2008) Practical poissonian-gaussian noise modeling and fitting for single-image raw-data. IEEE Transactions on Image Processing 17(10):1737–1754

Fu Y, Zhang T, Wang L, Huang H (2021) Coded hyperspectral image reconstruction using deep external and internal learning. IEEE Transactions Pattern Analysis and Machine Intelligence 44(7):3404–3420

Fu Y, Hong Y, Chen L, You S (2022) Le-gan: Unsupervised low-light image enhancement network using attention module and identity invariant loss. Knowledge-Based Systems 240:108010

Gatys LA, Ecker AS, Bethge M (2016) Image style transfer using convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2414–2423

Gnanasambandam A, Chan SH (2020) Image classification in the dark using quanta image sensors. In: Proceedings of European Conference on Computer Vision, pp 484–501

Gonzalez RC, Woods RE, et al. (2002) Digital image processing

Gu S, Li Y, Gool LV, Timofte R (2019) Self-guided network for fast image denoising. In: Proceedings of IEEE International Conference on Computer Vision, pp 2511–2520

Guo C, Li C, Guo J, Loy CC, Hou J, Kwong S, Cong R (2020) Zero-reference deep curve estimation for low-light image enhancement.

In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 1780–1789

Hahn J, Tai XC, Borok S, Bruckstein AM (2011) Orientation-matching minimization for image denoising and inpainting. International journal of computer vision 92(3):308–324

Hajiaboli MR (2011) An anisotropic fourth-order diffusion filter for image noise removal. International Journal of Computer Vision 92(2):177–191

He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 770–778

He K, Gkioxari G, Dollár P, Girshick R (2017) Mask r-cnn. In: Proceedings of IEEE International Conference on Computer Vision, pp 2961–2969

Hinton G, Vinyals O, Dean J (2015) Distilling the knowledge in a neural network. arXiv preprint arXiv:150302531 2(7)

Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 7132–7141

Huang Z, Huang L, Gong Y, Huang C, Wang X (2019) Mask scoring r-cnn. In: Proceedings of IEEE International Conference on Computer Vision, pp 6409–6418

Jiang Y, Gong X, Liu D, Cheng Y, Fang C, Shen X, Yang J, Zhou P, Wang Z (2021) Enlightengan: Deep light enhancement without paired supervision. IEEE Transactions on Image Processing 30(1):2340–2349

Julca-Aguilar F, Taylor J, Bijelic M, Mannan F, Tseng E, Heide F (2021) Gated3d: Monocular 3d object detection from temporal illumination cues. In: Proceedings of IEEE International Conference on Computer Vision, pp 2938–2948

Kirillov A, Wu Y, He K, Girshick R (2020) Pointrend: Image segmentation as rendering. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 9799–9808

Lamba M, Mitra K (2021) Restoring extremely dark images in real time. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 3487–3497

Lee Y, Park J (2019) Centermask: Real-time anchor-free instance segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 13906–13915

Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014a) Microsoft coco: Common objects in context. In: Proceedings of European Conference on Computer Vision, pp 740–755

Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014b) Microsoft coco: Common objects in context. In: Proceedings of European Conference on Computer Vision, pp 740–755

Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S (2017a) Feature pyramid networks for object detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 2117–2125

Lin TY, Goyal P, Girshick R, He K, Dollár P (2017b) Focal loss for dense object detection. In: Proceedings of IEEE International Conference on Computer Vision, pp 2980–2988

Liu D, Wen B, Jiao J, Liu X, Wang Z, Huang TS (2020a) Connecting image denoising and high-level vision tasks via deep learning. IEEE TIP 29(1):3695–3706

Liu J, Xu D, Yang W, Fan M, Huang H (2021a) Benchmarking low-light image enhancement and beyond. International Journal of Computer Vision 129(4):1153–1184

Liu L, Ouyang W, Wang X, Fieguth P, Chen J, Liu X, Pietikäinen M (2020b) Deep learning for generic object detection: A survey. International Journal of Computer Vision 128(2):261–318

Liu Y, Qin Z, Anwar S, Ji P, Kim D, Caldwell S, Gedeon T (2021b) Invertible denoising network: A light solution for real noise removal. In: Proceedings of IEEE Conference on Computer Vision and Pat-

tern Recognition, pp 13365–13374

Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021c) Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of IEEE International Conference on Computer Vision, pp 10012–10022

Liu Z, Mao H, Wu CY, Feichtenhofer C, Darrell T, Xie S (2022) A convnet for the 2020s. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 11976–11986

Loh YP, Chan CS (2019) Getting to know low-light images with the exclusively dark dataset. Computer Vision and Image Understanding 178(1):30–42

Lore KG, Akintayo A, Sarkar S (2017) Llnet: A deep autoencoder approach to natural low-light image enhancement. Pattern Recognition 61:650–662

Lv F, Li Y, Lu F (2021) Attention guided low-light image enhancement with a large scale low-light simulation dataset. International Journal of Computer Vision 129(7):2175–2193

Mohan R, Valada A (2021) Efficientps: Efficient panoptic segmentation. International Journal of Computer Vision 129(5):1551–1579

Morawski I, Chen YA, Lin YS, Hsu WH (2021) Nod: Taking a closer look at detection under extreme low-light conditions with night object detection dataset. In: Proceedings of the British Machine Vision Conference, pp 1–13

Plotz T, Roth S (2017) Benchmarking denoising algorithms with real photographs. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 1586–1595

Punnappurath A, Abuolaim A, Abdelhamed A, Levinshtein A, Brown MS (2022) Day-to-night image synthesis for training nighttime neural isps. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 10769–10778

Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: Unified, real-time object detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 779–788

Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: Towards real-time object detection with region proposal networks. In: Proceedings of Advances in Neural Information Processing Systems, pp 91–99

Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention, pp 234–241

Sakaridis C, Dai D, Van Gool L (2018) Semantic foggy scene understanding with synthetic data. International Journal of Computer Vision 126(9):973–992

Sakaridis C, Dai D, Gool LV (2019) Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In: Proceedings of IEEE International Conference on Computer Vision, pp 7374–7383

Sasagawa Y, Nagahara H (2020) Yolo in the dark-domain adaptation method for merging multiple models. In: Proceedings of European Conference on Computer Vision, pp 345–359

Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I, Fergus R (2014) Intriguing properties of neural networks. In: Proceedings of International Conference on Learning Representations, pp 1–10

Tan S, Jiao L (2007) Multivariate statistical models for image denoising in the wavelet domain. International Journal of Computer Vision 75(2):209–230

Tan X, Xu K, Cao Y, Zhang Y, Ma L, Lau RW (2021) Night-time scene parsing with a large real dataset. IEEE Transactions on Image Processing 30(1):9085–9098

Tian Z, Shen C, Chen H, He T (2019) Fcos: Fully convolutional one-stage object detection. In: Proceedings of IEEE International Conference on Computer Vision, pp 9627–9636

Ulyanov D, Vedaldi A, Lempitsky V (2020) Deep image prior. International Journal of Computer Vision 128(7):1867–1889

Wang W, Wei C, Yang W, Liu J (2018a) Gladnet: Low-light enhancement network with global awareness. In: Proceedings of IEEE International Conference on Automatic Face & Gesture Recognition, pp 751–755

Wang W, Yang W, Liu J (2021) Hla-face: Joint high-low adaptation for low light face detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 16195–16204

Wang X, Girshick R, Gupta A, He K (2018b) Non-local neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7794–7803

Wei C, Wang W, Yang W, Liu J (2018) Deep retinex decomposition for low-light enhancement. In: Proceedings of the British Machine Vision Conference, pp 1–12

Wei K, Fu Y, Yang J, Huang H (2020) A physics-based noise formation model for extreme low-light raw denoising. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 2758–2767

Wei K, Fu Y, Zheng Y, Yang J (2021) Physics-based noise modeling for extreme low-light photography. IEEE Transactions on Pattern Analysis and Machine Intelligence 1(1):1–17

Woo S, Park J, Lee JY, Kweon IS (2018) Cbam: Convolutional block attention module. In: Proceedings of European Conference on Computer Vision, pp 3–19

Xiang Y, Fu Y, Zhang L, Huang H (2019) An effective network with convlstm for low-light image enhancement. In: Pattern Recognition and Computer Vision, pp 221–233

Xie C, Wu Y, Maaten Lvd, Yuille AL, He K (2019) Feature denoising for improving adversarial robustness. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 501–509

Xu K, Yang X, Yin B, Lau RW (2020) Learning to restore low-light images via decomposition-and-enhancement. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 2281–2290

Yang H, Kaixuan W, Linwei C, Ying F (2021) Crafting object detection in very low light. In: Proceedings of the British Machine Vision Conference, pp 1–15

Yang W, Yuan Y, Ren W, Liu J, Scheirer WJ, Wang Z, Zhang T, Zhong Q, Xie D, Pu S, et al. (2020a) Advancing image understanding in poor visibility environments: A collective benchmark study. IEEE Transactions on Image Processing 29(1):5737–5752

Yang W, Yuan Y, Ren W, Liu J, Scheirer WJ, Wang Z, Zhang T, Zhong Q, Xie D, Pu S, et al. (2020b) Advancing image understanding in poor visibility environments: A collective benchmark study. IEEE TIP 29(1):5737–5752

Zhang F, Li Y, You S, Fu Y (2021a) Learning temporal consistency for low light video enhancement from single images. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp 4967–4976

Zhang T, Fu Y, Zhang J (2022) Guided hyperspectral image denoising with realistic data. International Journal of Computer Vision 130(11):2885–2901

Zhang Y, Guo X, Ma J, Liu W, Zhang J (2021b) Beyond brightening low-light images. International Journal of Computer Vision 129(4):1013–1037