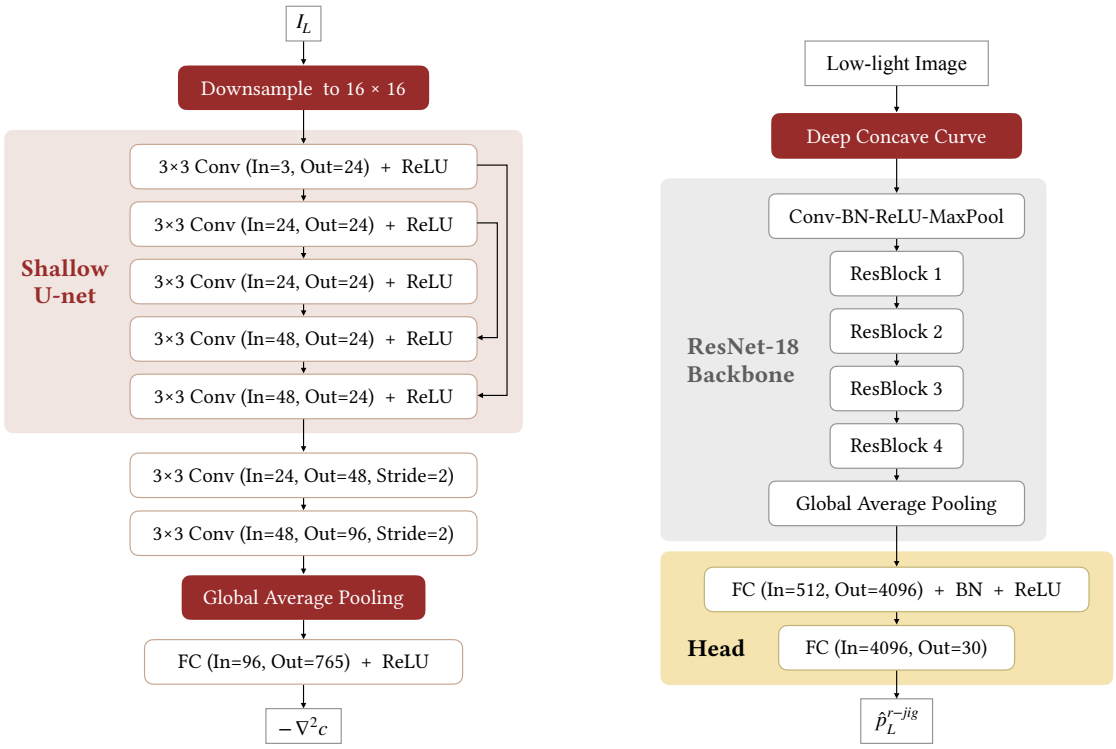
A APPENDIX

A.1 Implementation: Deep Concave Curve

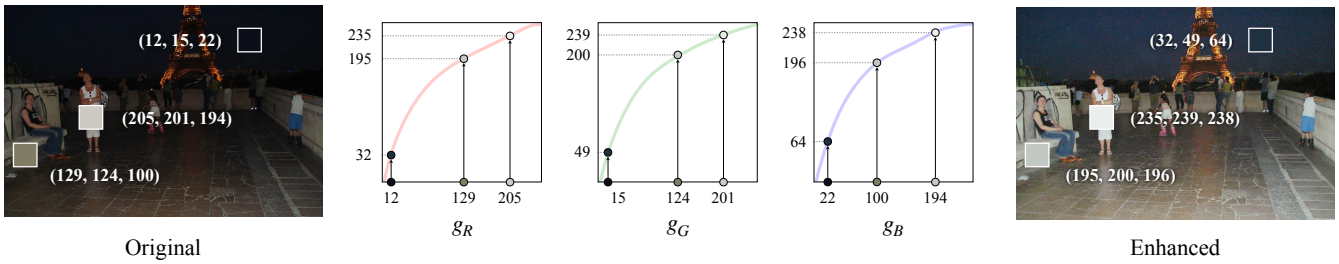
The detailed network architecture of our curve predictor is shown in Fig. A1(a). We first downsample the input image I_L to 16×16 resolution by the nearest neighbor algorithm. Then, the downsampled image is fed to a shallow U-net [18], which consists of five standard convolution layers without down-sampling and up-sampling. Each convolution layer is followed by a Rectified Linear Unit (ReLU). After the shallow U-net, two convolution layers downsample the feature by stride 2. Finally, we use global average pooling and a fully connected layer to map the feature to 765 channels.



(a) Curve predictor (b) Framework for classification

Figure A1: The architecture of our curve predictor and our framework for classification.

We provide an illustration of how to apply predicted deep concave curve g to images in Fig. A2. Our g defines that for each original pixel value, what its new pixel value is. Considering 8-bit, g is a vector of length 256. For original pixels of value $(p-1)/256$ on the input image, where $p \in \{1, 2, .., 256\}$, their new pixel values are the p-th element of g. Considering RGB, we predict an independent g for each color channel, *i.e.*, we have g_R, g_G, g_B.



Original Enhanced

Figure A2: How to adjust an image with the given deep concave curve.

A.2 Implementation: Low-light Object Classification

For adaptive low-light classification, the detailed framework is shown in Fig. A1(b). Our baseline is ResNet-18 [8]. The rotated jigsaw pretext task head is added to the feature after the adaptive average pooling and before the last fully connected layer. The head consists of two fully connected layers with one batch norm.

In asymmetric cross-domain self-supervised training, we put 3×3 patches together into a single image and set the permutation number to 30. In other words, the proposed rotated jigsaw is a 30-category classification task. Following [16], we choose permutations with sufficiently large average Hamming distance.

For SACC, we first pretrain the pretext task head on normal light images with a learning rate of 0.01, which is multiplied by 0.1 at the 150,000th iteration. Pretraining lasts for 300,000 iterations until $\mathcal{L}_N$ and $\mathcal{L}_L$ do not decrease. Note that although the model is pretrained on normal light images, we still compute $\mathcal{L}_L$ on low-light images (without back-propagation) to better control the pretraining process. After pretraining, we fix the pretext task head and train our deep concave curve for 20,000 iterations with a learning rate of 0.01, which degrades to 0.001 at the 5,000th iteration and finally degrades to 0.0001 at the 10,000th iteration. The batch sizes are all set to 64. The optimizer is SGD [17], with a momentum of 0.9 and a weight decay of 0.00001. Random cropping is used for data augmentation.

For SACC+, we first use trained SACC to predict pseudo labels for the low-light dataset. We filter predictions with a confidence threshold of 0.98. Then we fix the deep concave curve and fine-tune ResNet-18 with low-light pseudo labels and normal light ground truth labels. Fine-tuning takes 7000 iterations with an initial learning rate of 0.001, which degrades to 0.1× after 2000, 4000, and 6000 iterations. The mini-batch size is 64. The optimizer is SGD [17], with a momentum of 0.9 and a weight decay of 0.00001. Random cropping, horizontal flipping, color jittering, and random rotation are used for data augmentation.

A.3 Implementation: Dark Face Detection

For adaptive dark face detection, the detailed framework is shown in Fig. A3(a). Our baseline is DSFD [14], which has a VGG [19] backbone. We use the input features of its first shot detection stage, *i.e.*, conv3_3, conv4_3, conv5_3, conv_fc7, conv6_2, and conv7_2. Each pretext task head consists of two convolutional layers and one fully-connected layer.

For SACC, we first pretrain the head on normal light images under the task of original jigsaw permutation for 300,000 iterations with a learning rate of 0.001, then under the task of rotated jigsaw permutation for 200,000 iterations with the same learning rate. In this way, the pretext task can fully converge. After pretraining, we fix the head and train our deep concave curve for 25,000 iterations with an initial learning rate of 0.0002, which degrades to 0.5× after 20,000 iterations. The batch sizes are all set to 32. We use Adam [11] for optimization, with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and no weight decay L2 penalty. Random cropping is used for data augmentation.

For SACC+, we also first use trained SACC to predict pseudo labels for the low-light dataset. We filter predictions with a confidence threshold of 0.4. Then, we fine-tune DSFD with mini-batch size 8 for 100,000 iterations. The initial learning rate is 0.0001 and multiplied by 0.1 after 50,000 iterations. The optimizer is SGD [17], with a momentum of 0.9 and a weight decay of 0.0005. We use two data augmentation strategies: random cropping and color jittering. We follow Sec. A.2 for other details.

A.4 Implementation: Low-light Action Recognition

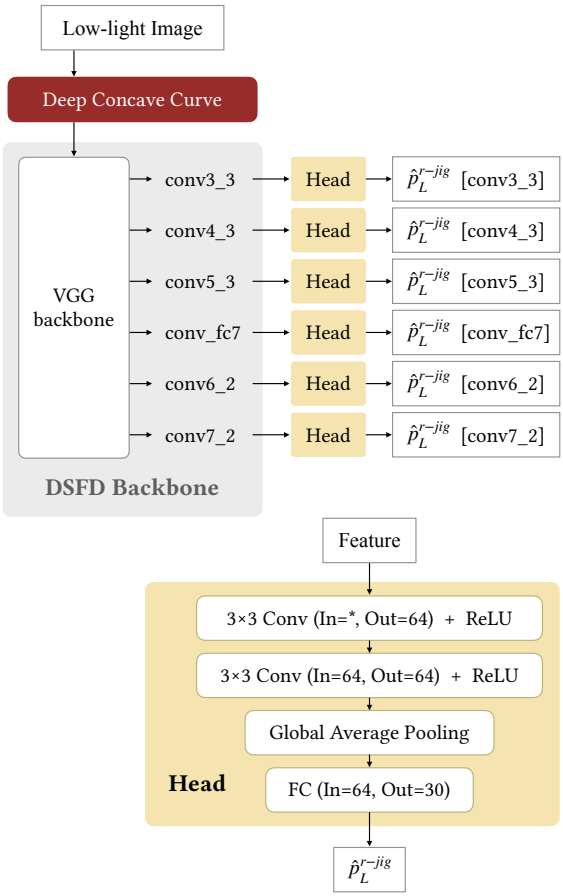
For low-light video action recognition, the detailed framework is shown in Fig. A3(b). The baseline classifier is I3D [2], which is based on 3D-ResNet [3]. The pretext task head is fed with the feature after the adaptive average pooling and before the last fully connected layer. It consists of a dropout and a fully connected layer.

We first pretrain the pretext task head on normal light images with a learning rate of 0.001 for 100,000 iterations, which is multiplied by 0.1 after 50,000 iterations. After pretraining, we fix the pretext task head and train our deep concave curve for 30,000 iterations with a learning rate of 0.0001. The mini-batch size is 4. The optimizer is SGD [17], with a momentum of 0.9 and a weight decay of 0.00001. Other details are the same as Sec. A.2.

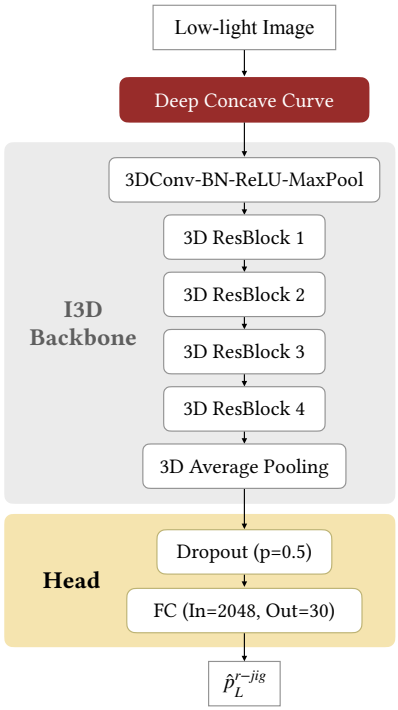
A.5 Implementation: Optical Flow Estimation in the Dark

For low-light optical flow estimation, the detailed framework is shown in Fig. A3(c). The baseline is PWC-Net [20]. We use its 6-layer feature pyramid. Each pretext task head consists of two convolutional layers and one fully-connected layer.

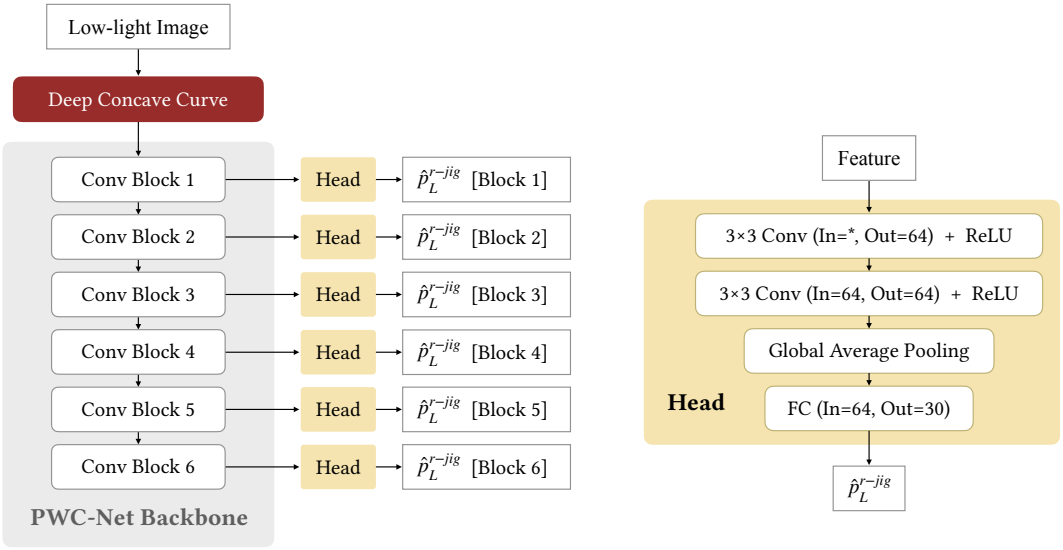
We first pretrain the pretext task head on normal light images, then fix the pretext task head and train our deep concave curve on low-light images. For both two processes, the models are trained for 60,000 iterations, with an initial learning rate of 0.001 and multiplied by 0.1 after 50,000 iterations. The mini-batch size is 64 for pretext task pretraining and 32 for training deep concave curve. The optimizer is SGD [17], with a momentum of 0.9 and a weight decay of 0.00001. We follow Sec. A.2 for other settings.



(a) Framework for face detection



(b) Framework for video action recognition



(c) Framework for optical flow estimation

Figure A3: The architecture of our framework for face detection, action recognition, and optical flow estimation.

A.6 Benchmarking Settings

The code sources of all compared methods are shown in Table A1. Thanks to the authors for sharing their codes, which are very helpful for our research work. All methods use the same backbone and normal light pretraining. We follow CIConv [12] and HLA-Face [21] for other benchmarking details.

Table A1: Code sources of compared methods.

Method	Link
RetinexNet [23]	https://github.com/weichen582/RetinexNet
KinD [26]	https://github.com/zhangyhuaee/KinD
LIME [7]	https://sites.google.com/view/xjguo/lime
MF [4]	https://github.com/baidut/BIMEF
SMOID [9]	https://github.com/MichaelHYJiang/Learning-to-See-Moving-Objects-in-the-Dark
EnlightenGAN [10]	https://github.com/VITA-Group/EnlightenGAN
Zero-DCE [6]	https://github.com/Li-Chongyi/Zero-DCE
Zero-DCE++ [13]	https://github.com/Li-Chongyi/Zero-DCE_extension
RUAS [15]	https://github.com/KarelZhang/RUAS
StableLLVE [25]	https://github.com/zkawfanx/StableLLVE
LLFlow [22]	https://github.com/wyf0912/LLFlow
CMD [24]	https://gist.github.com/yusuke0519/724aa68fc431afadb0cc7280168da17b
MMD [1]	https://github.com/jindongwang/transferlearning/
DANN [5]	https://github.com/fungtion/DANN
HLA-Face [21]	https://github.com/daooshee/HLA-Face-Code
CIConv [12]	https://github.com/Attila94/CIConv
Zheng *et al.* [27]	https://github.com/mf-zhang/Optical-Flow-in-the-Dark/tree/main/PWCNet_OFDark

A.7 Computational Complexity and Running Time Analysis

We show the computational complexity and running time of low-light enhancement methods in Table A2. The performance is tested on RGB images of resolution 1024×1024, with Intel i7-9700 K @3.60 GHz and GeForce GTX 1080. For running time analysis, we report the average time for 100 images.

Our model is very lightweight. Among all deep low-light enhancement models, our deep concave curve has the lowest computational complexity and the second shortest average running time. Compared with Zero-DCE [6], our model has more learnable parameters but less computational complexity and running time, indicating that our deep concave curve is more powerful while being more efficient.

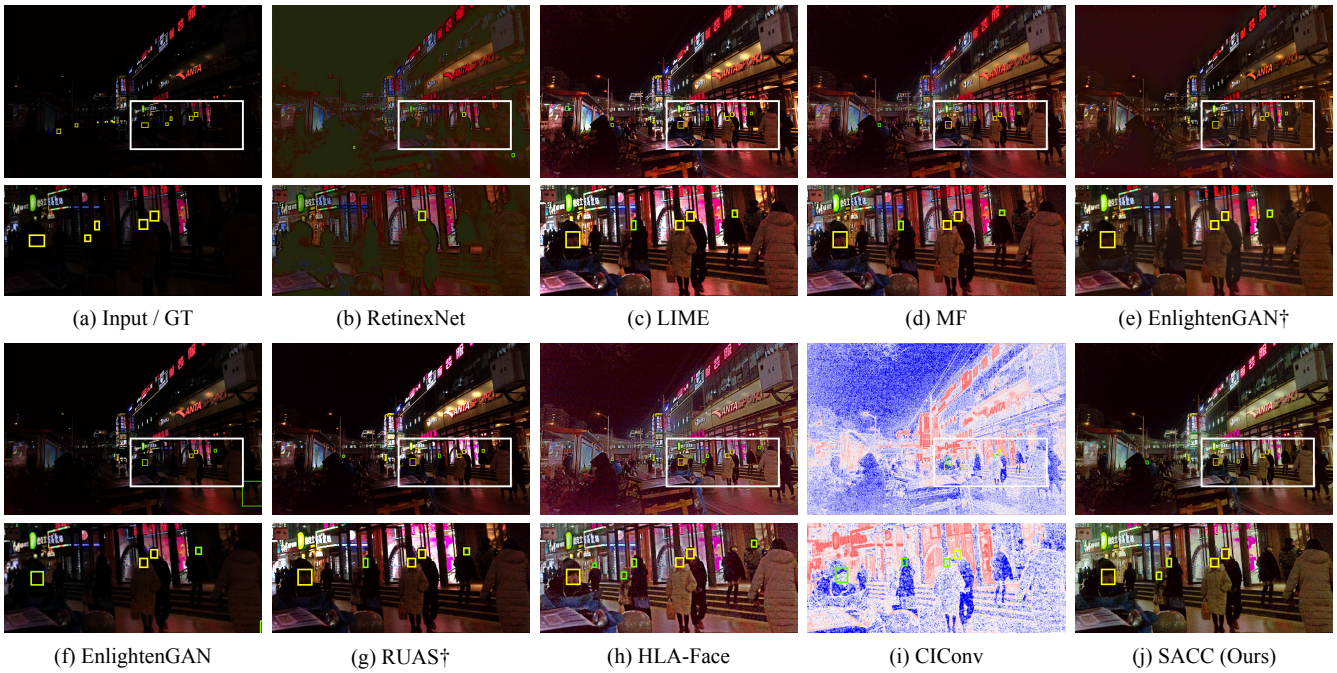
By merging multiple frames and predicting a unified g for the whole video sequence, our model is faster on videos as shown in Table A2. For 1920×1080 videos, the average running time of SACC-Video is only 13.7 ms, supporting 1080p real-time enhancement.

Table A2: Comparison of computational complexity (FLOPs) and running time.

Method	FLOPs (GMac)	Params	Time (ms)
LLFlow [22]	4588.72	38.86 M	2663.9
RetinexNet [23]	324.04	597.7 k	133.2
EnlightenGAN [10]	263.31	8.64 M	97.3
Zero-DCE [6]	83.48	79.42 k	50.9
RUAS [15]	3.80	3.44 k	56.4
Zero-DCE++ [13]	0.08	10.56 k	5.8
SACC (Ours)	0.01	158.06 k	19.5
SACC-Video (Ours)			7.1

A.8 More Subjective Results for Dark Face Detection

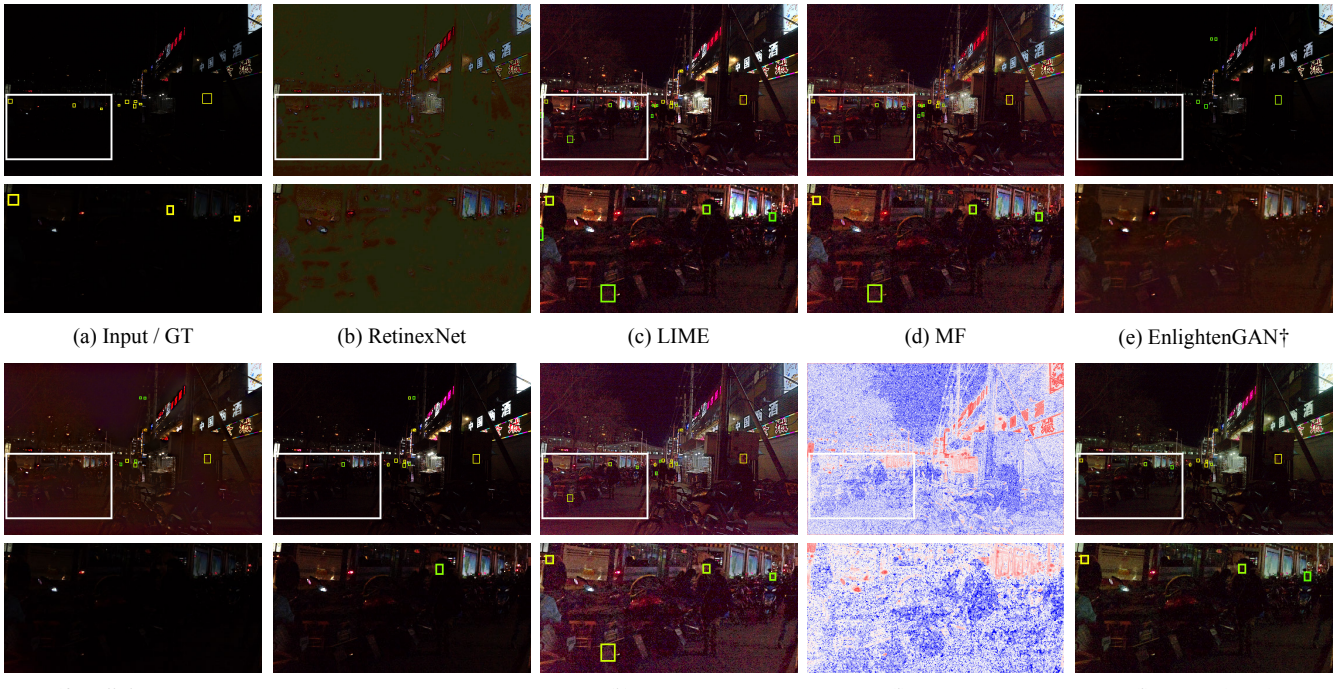
More subjective results can be found in Figs. A5-A8. Compared with other methods, our SACC can better recognize faces and less misidentify faces from non-face objects. Two sub-figures in Fig. 6 (main paper) miss bounding boxes. We correct them in Fig. A4.



(a) Input / GT (b) RetinexNet (c) LIME (d) MF (e) EnlightenGAN†

(f) EnlightenGAN (g) RUAS† (h) HLA-Face (i) CIConv (j) SACC (Ours)

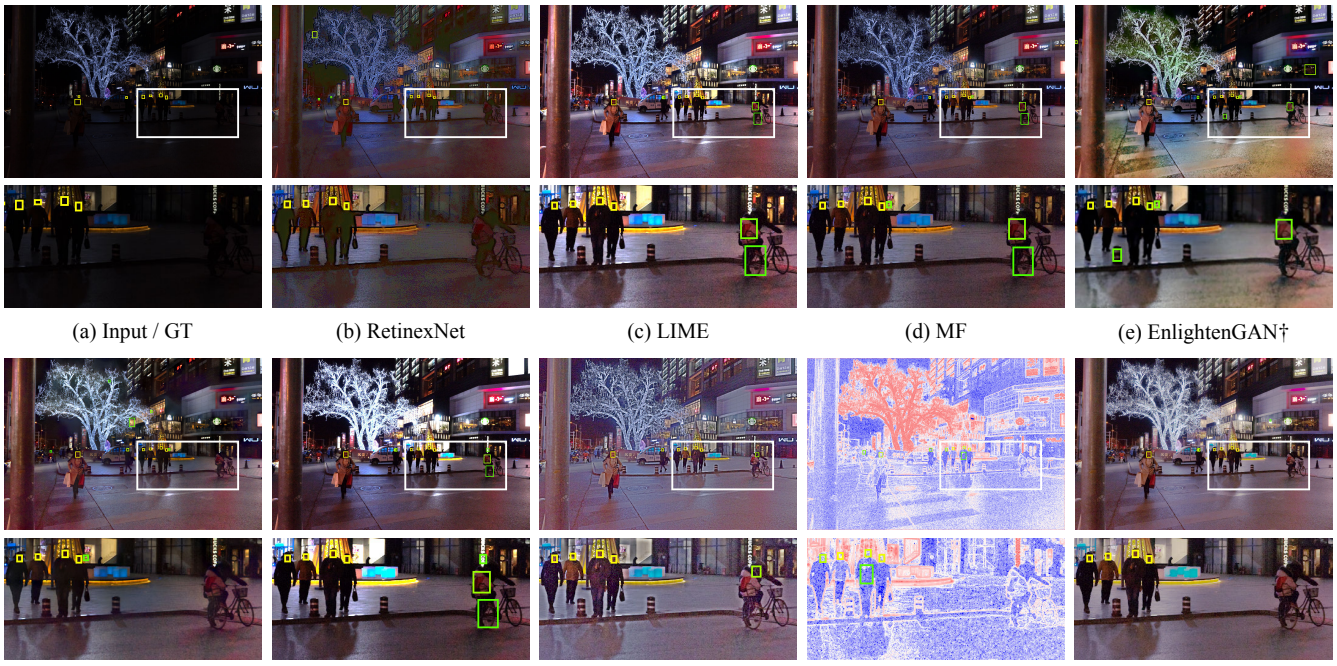
Figure A4: More subjective comparison results for dark face detection. † denotes that the low-light enhancement model is retrained. The color of the bounding boxes represents the confidence of recognition, with yellow indicating higher confidence.



(a) Input / GT (b) RetinexNet (c) LIME (d) MF (e) EnlightenGAN†

(f) EnlightenGAN (g) RUAS† (h) HLA-Face (i) CIConv (j) SACC (Ours)

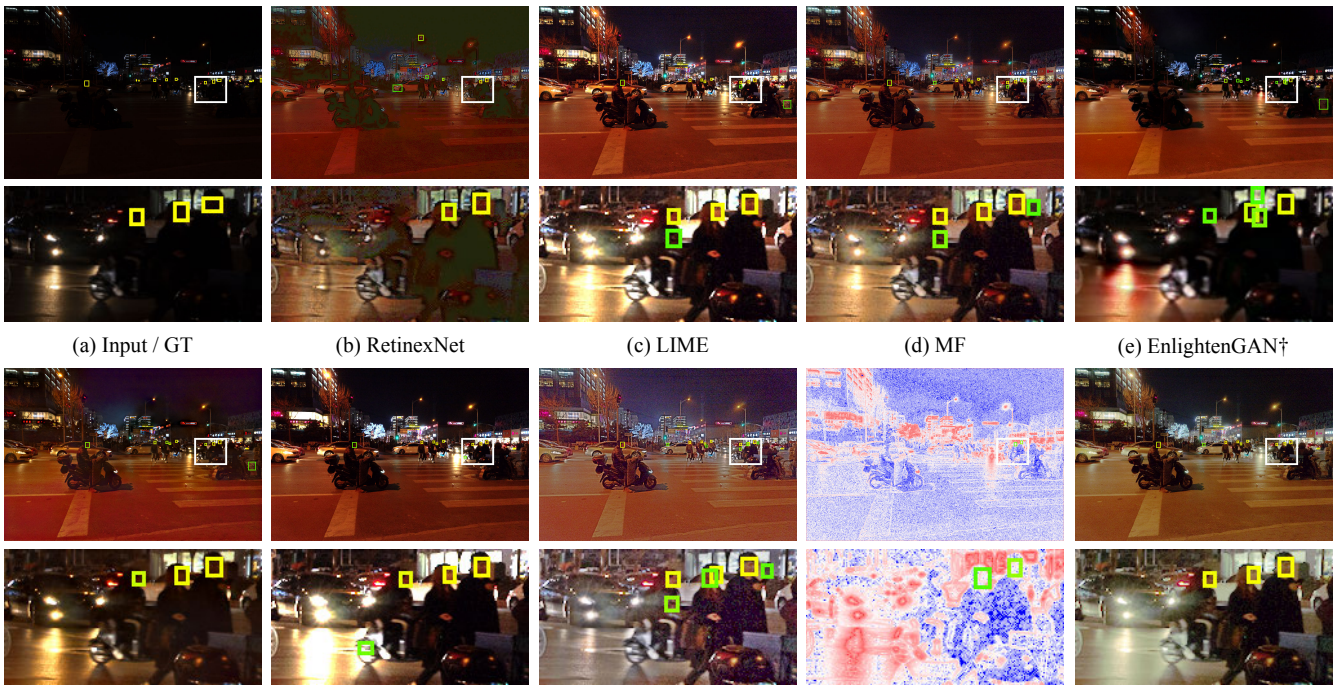
Figure A5: More subjective comparison results for dark face detection. † denotes that the low-light enhancement model is retrained. The color of the bounding boxes represents the confidence of recognition, with yellow indicating higher confidence.



(a) Input / GT (b) RetinexNet (c) LIME (d) MF (e) EnlightenGAN†

(f) EnlightenGAN (g) RUAS† (h) HLA-Face (i) CIConv (j) SACC (Ours)

Figure A6: More subjective comparison results for dark face detection. † denotes that the low-light enhancement model is retrained. The color of the bounding boxes represents the confidence of recognition, with yellow indicating higher confidence.



(a) Input / GT (b) RetinexNet (c) LIME (d) MF (e) EnlightenGAN†

(f) EnlightenGAN (g) RUAS† (h) HLA-Face (i) CIConv (j) SACC (Ours)

Figure A7: More subjective comparison results for dark face detection. † denotes that the low-light enhancement model is retrained. The color of the bounding boxes represents the confidence of recognition, with yellow indicating higher confidence.

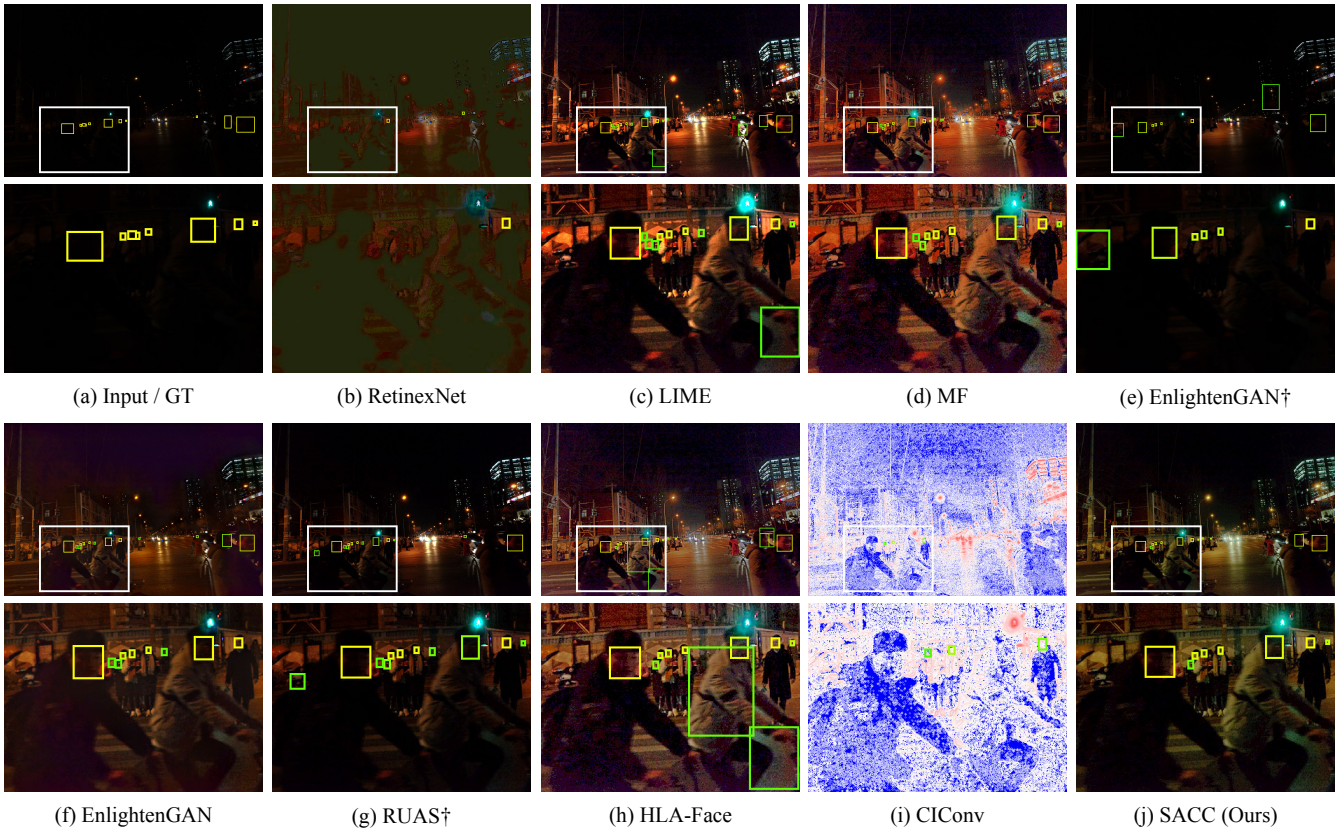


Figure A8: More subjective comparison results for dark face detection. † denotes that the low-light enhancement model is retrained. The color of the bounding boxes represents the confidence of recognition, with yellow indicating higher confidence.

A comparison between our SACC and SACC+ can be found in Fig. A9. We show three groups of results. The left three images are results under extremely low illumination (average pixel value < 5). Compared with SACC, SACC+ can better handle these hard cases. The middle three images show results where SACC misidentifies non-face objects, while SACC+ is more robust. The right three images show results where SACC+ can better handle small faces and different face angles.

Figure A9: Comparison between our SACC and our advanced version SACC+.

A.9 Subjective Results for Video Action Recognition

Fig. A10 is a video about waving hands. Before enhancement, the prediction is "Drink" with confidence 66%. After enhanced by our SACC, the prediction is "Wave" with confidence 94%.

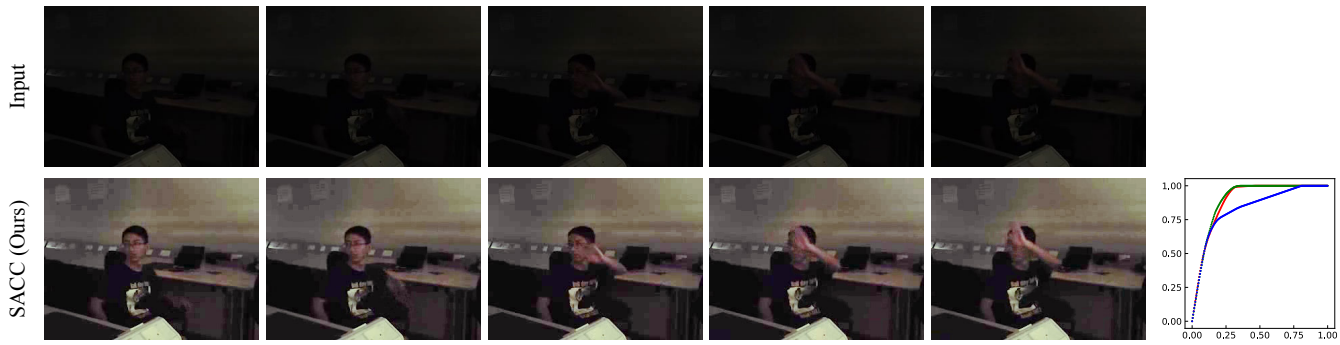


Figure A10: Subjective results for video action recognition.

REFERENCES

[1] Karsten M. Borgwardt, Arthur Gretton, Malte J. Rasch, Hans-Peter Kriegel, Bernhard Schölkopf, and Alexander J. Smola. 2006. Integrating Structured Biological Data by Kernel Maximum Mean Discrepancy. In *ISMB*.
[2] João Carreira and Andrew Zisserman. 2017. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In *CVPR*.
[3] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. 2019. SlowFast Networks for Video Recognition. In *ICCV*.
[4] Xueyang Fu, Delu Zeng, Yue Huang, Yinghao Liao, Xinghao Ding, and John W. Paisley. 2016. A Fusion-Based Enhancing Method for Weakly Illuminated Images. *Signal Processing* 129 (2016), 82–96.
[5] Yaroslav Ganin and Victor S. Lempitsky. 2015. Unsupervised Domain Adaptation by Backpropagation. In *ICML*.
[6] Chunle Guo, Chongyi Li, Jichang Guo, Chen Change Loy, Junhui Hou, Sam Kwong, and Runmin Cong. 2020. Zero-Reference Deep Curve Estimation for Low-Light Image Enhancement. In *CVPR*.
[7] Xiaojie Guo, Yu Li, and Haibin Ling. 2017. LIME: Low-Light Image Enhancement via Illumination Map Estimation. *IEEE TIP* 26, 2 (2017), 982–993.
[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *CVPR*.
[9] Haiyang Jiang and Yinqiang Zheng. 2019. Learning to See Moving Objects in the Dark. In *ICCV*.
[10] Yifan Jiang, Xinyu Gong, Ding Liu, Yu Cheng, Chen Fang, Xiaohui Shen, Jianchao Yang, Pan Zhou, and Zhangyang Wang. 2021. EnlightenGAN: Deep Light Enhancement Without Paired Supervision. *IEEE TIP* 30 (2021), 2340–2349.
[11] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.
[12] Attila Lengyel, Sourav Garg, Michael Milford, and Jan C. van Gemert. 2021. Zero-Shot Domain Adaptation with a Physics Prior. In *ICCV*.
[13] Chongyi Li, Chunle Guo, and Chen Change Loy. 2021. Learning to Enhance Low-Light Image via Zero-Reference Deep Curve Estimation. *IEEE TPAMI* (2021).
[14] Jian Li, Yabiao Wang, Changan Wang, Ying Tai, Jianjun Qian, Jian Yang, Chengjie Wang, Jilin Li, and Feiyue Huang. 2019. DSFD: Dual Shot Face Detector. In *CVPR*.
[15] Risheng Liu, Long Ma, Jiaao Zhang, Xin Fan, and Zhongxuan Luo. 2021. Retinex-Inspired Unrolling With Cooperative Prior Architecture Search for Low-Light Image Enhancement. In *CVPR*.
[16] Mehdi Noroozi and Paolo Favaro. 2016. Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles. In *ECCV*.
[17] Herbert Robbins and Sutton Monro. 1951. A Stochastic Approximation Method. *The Annals of Mathematical Statistics* 22, 3 (1951), 400–407.
[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *MICCAI*.
[19] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *ICLR*.
[20] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. 2018. PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume. In *CVPR*.
[21] Wenjing Wang, Wenhan Yang, and Jiaying Liu. 2021. HLA-Face: Joint High-Low Adaptation for Low Light Face Detection. In *CVPR*.
[22] Yufei Wang, Renjie Wan, Wenhan Yang, Haoliang Li, Lap-Pui Chau, and Alex C. Kot. 2022. Low-Light Image Enhancement with Normalizing Flow. In *AAAI*.
[23] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. 2018. Deep Retinex Decomposition for Low-Light Enhancement. In *BMVC*.
[24] Werner Zellinger, Thomas Grubinger, Edwin Lughofer, Thomas Natschläger, and Susanne Saminger-Platz. 2017. Central Moment Discrepancy (CMD) for Domain-Invariant Representation Learning. In *ICLR*.
[25] Fan Zhang, Yu Li, Shaodi You, and Ying Fu. 2021. Learning Temporal Consistency for Low Light Video Enhancement From Single Images. In *CVPR*.
[26] Yonghua Zhang, Jiawan Zhang, and Xiaojie Guo. 2019. Kindling the Darkness: A Practical Low-light Image Enhancer. In *ACM MM*.
[27] Yinqiang Zheng, Mingfang Zhang, and Feng Lu. 2020. Optical Flow in the Dark. In *CVPR*.