SYENet: A Simple Yet Effective Network for Multiple Low-Level Vision Tasks with Real-time Performance on Mobile Device

Weiran Gou*[1,2], Ziyao Yi*[1,2], Yan Xiang*[1,2], Shaoqing Li[1,2], Zibin Liu[1,2], Dehui Kong[1,2], and Ke Xu†[1,2]

[1]State Key Laboratory of Mobile Network and Mobile Multimedia Technology
[2]Sanechips Technology, Chengdu, China
{gou.weiran, yi.ziyao, xiang.yan1, li.shaoqing1, liu.zibin, kong.dehui, xu.kevin}@sanechips.com.cn
https://github.com/sanechips-multimedia/syenet

Abstract

*With the rapid development of AI hardware accelerators, applying deep learning-based algorithms to solve various low-level vision tasks on mobile devices has gradually become possible. However, two main problems still need to be solved: task-specific algorithms make it difficult to integrate them into a single neural network architecture, and large amounts of parameters make it difficult to achieve real-time inference. To tackle these problems, we propose a novel network, SYENet, with only 6K parameters, to handle multiple low-level vision tasks on mobile devices in a real-time manner. The SYENet consists of two asymmetrical branches with simple building blocks. To effectively connect the results by asymmetrical branches, a Quadratic Connection Unit(**QCU**) is proposed. Furthermore, to improve performance, a new Outlier-Aware Loss is proposed to process the image. The proposed method proves its superior performance with the best PSNR as compared with other networks in real-time applications such as Image Signal Processing(ISP), Low-Light Enhancement(LLE), and Super-Resolution(SR) with 2K60FPS throughput on Qualcomm 8 Gen 1 mobile SoC(System-on-Chip). Particularly, for ISP task, SYENet got the highest score in MAI 2022 Learned Smartphone ISP challenge.*

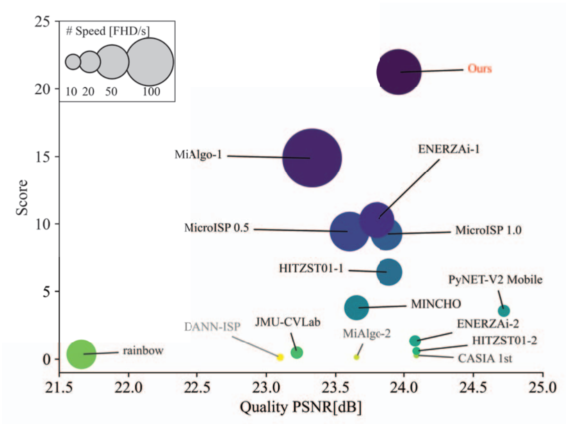
1. Introduction

In recent years, with the thriving development of AI accelerators [54, 77], such as Neural Processor Units(NPUs) or Graphic Processor Units(GPUs), AI algorithms can be deployed on mobile devices and achieved great success [65, 73, 94, 99]. Many mobile SoCs, especially those designed for smartphone, tablet, and in-vehicle infotainment systems, require superior visual quality processing, which cannot be achieved without leveraging deep networks such as ISP [43, 46], LLE [7], and SR [8, 11, 14]. However, due to the tight hardware constraints such as power and computing resources, deploying these algorithms on mobile devices still has several issues as follows.

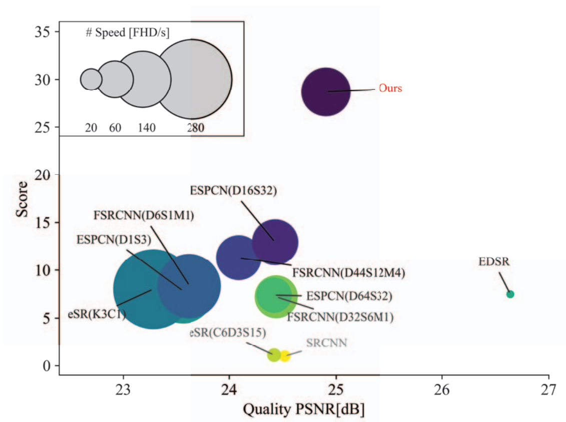
The first issue concerns real-time processing. Usually, these low-level vision tasks require a 2K60FPS or even higher real-time performance to satisfy the viewer's needs. Although the State-of-the-Arts(SOTAs) [8, 14, 46, 90] dealing with similar tasks have boosted the performance, they increased the numbers of parameters and computational cost drastically, which cannot satisfy real-time inference deployment even on powerful hardware such as server-level processors. Moreover, compared with high-level tasks [94, 99], where the input images could be resized into a lower resolution such as 128×128 or 256×256 without noticeable effects, low-level vision tasks cannot do the same thing as their preliminary goal is to improve the human visual quality. A more detailed discussion about the constraints of low-level vision tasks is in Appendix G.

The second issue is related to hardware resources on mobile devices such as Qualcomm's Snapdragon. As compared with server-level Central Processing Unit(CPU) or GPU, mobile SoC usually has limited computing resources such as multiplication-and-accumulation units, limited memory bandwidth, and limited power consumption budget. Unfortunately, most low-level vision algorithms are task-specific [7, 14, 46, 62] and independent to each other, which makes it difficult to merge into a single architecture. To make things worse, many advanced operators, such as deformable convolution [106] and 3D-convolution [68], cannot be directly applied on mobile devices, which further

*These authors contributed equally to this work.
†Corresponding author: Ke Xu(xu.kevin@sanechips.com.cn)



(a) ISP



(b) SR with scale factor $\times 4$

Figure 1: Comparison about different issues (a)ISP (b)SR$\times$4 upon comprehensive score versus quantitative measurements by SOTA models. The size of the model represents the inference speed. The Score equation is in Eq. 9 by the MAI Challenge [43]. Our method shows superior comprehensive performance upon image quality, inference speed, and the score involving both factors.

leads to performance degradation. Therefore, as already proved in high-level vision tasks and NLP [4, 10, 31, 33], building a simple yet unified network architecture is the best choice for low-level vision tasks running on limited computing resources. Although there are excellent multiple low-level vision works like [8, 9, 60], they are not feasible for deployment on mobile devices due to their hardware complexity.

Several lightweight models [2, 57, 64, 89] were already proposed with a relatively small number of parameters to achieve a reasonable performance. Unfortunately, their implementations cannot satisfy real-time requirements such as 2K60FPS. To the best of our knowledge, there is still no prior work for the multiple low-level vision tasks in a single network architecture.

In this paper, we propose a new architecture SYENet, which can solve multiple low-level vision tasks with 2K60FPS on a mobile device such as Qualcomm's 8 Gen 1. We first decompose the low-level vision into two sub-tasks, which are texture generation and pattern classification. We then leverage two asymmetric branches to handle each task and a Quadratic Connection Unit(**QCU**) to connect the outputs to enlarge the representational power. Furthermore, the network replaces ordinary convolution with revised reparameterized convolution to boost the performance without increasing inference time, and Channel Attention(CA) is utilized for enhancement by global information. In addition, we propose Outlier-Aware Loss by involving global information and putting more focus on the outliers of the prediction for improving the performance. The proposed network achieves SOTA performance, as compared with other methods on low-level tasks. The comprehensive performance evaluation of SR, LLE and ISP tasks are shown in Table 1, 2, and 3, respectively.

The contributions of this paper can be summarized in three aspects:

1. We propose that asymmetric branches fused with Quadratic Connections Unit(**QCU**) is an effective method for solving multiple low-level vision tasks due to its ability to enlarge the representation power with modicum parameter count. Building upon this structure, we introduce SYENet, which incorporates revised reparameterized convolutions and channel attention to enhance performance without sacrificing speed.

2. A new loss function termed **Outlier-Aware Loss** is proposed for better training by leveraging global information and prioritizing outliers, the poorly predicted pixels.

3. Compared with other studies, our network has a superior performance according to the evaluation metrics in MAI Challenge [40], which reflects both the image quality and efficiency as shown in Fig. 1.

2. Related work

2.1. Low-level vision

Low-level vision techniques are generally required in a variety of applications to improve image and video quality. It could be defined as finding the best mapping between input and output images. In this section, we mainly discuss three widely used low-level vision tasks, which are super-resolution **SR**, end-to-end image signal processing **ISP**, and low-light enhancement **LLE**.

Super resolution: Convolution Neural Network(CNN) are widely used in SR algorithms. From the very first model

SRCNN [18] to EDSR [62], ESPCN [76], FEQE [83] and VDSR [52] .etc, CNNs significantly improve [14, 70, 102] SR performance and try to reduce the computational complexity. Special building blocks such as residual block [14, 34, 101] and deformable convolution [53, 88] are also used to improve visual quality. Transformer-based SR models such as SwinSR [60] and IPT [8] show significant improvements compared to traditional CNN-based models.

End-to-end ISP: HighEr-Resolution Network(HERN) [69] employs a two-branch structure to combine features of different scales to help conduct the tasks of demosaicing and image enhancement. PyNet [46] achieves similar performance as compared with the most sophisticated traditional ISP pipelines. AWNet [13] introduces attention mechanism, and wavelet transform for learning-based ISP network, which significantly improves image quality due to a large receptive field. Focusing on the color inconsistency issue that exists between input raw and output sRGB images, Zhang [105] designs a joint learning network. Similarly, from the perspective of solving noise discrepancy, Cao [5] introduces a pseudo-ISP, utilizing unpaired learning algorithm.

Low-light enhancement: Some end-to-end RAW-to-RGB LLE methods [23, 30, 75] employ the color shuffle operator in the front of the network. In the sRGB domain, with the advantage of being interpretable, many researchers focus on the decomposition method for LLE task, enhancing neural network designs and additional regularization as used in de-haze and de-noise [58, 61, 74, 86, 103]. Based on the non-local evaluation, normal light output can be obtained through a global awareness or generation method [49, 85, 87].

2.2. Mobile devices implementation

The SOTA networks for solving low-level vision problems show increasingly good performance. However, most of them are too computationally expensive, and hence it is tough to implement those algorithms in mobile devices without a powerful GPU. Meanwhile, some research about compact and effective network were carried out. Wang [89] proposed a lightweight U-shape network to support denoising operations on mobile platforms. MobiSR [57] with model compression methods applies two networks focusing on latency as well as quality to guarantee efficiency. SplitSR [64] reached 5 times faster inference using lightweight residual block, and XLSR [2] applies deep roots module [47] into SISR issue demonstrating the same performance of VDSR [52] using 30 times fewer parameters. Unfortunately, however, lightweight networks still preserve millions of parameters, which is far from the real-time application of 2K60FPS in mobile devices.

2.3. Re-parameterization

Re-parameterization is the approach for structural simplification using re-parameterized blocks, which is complicated during training but simplified during inference with the equivalent forward results. ACNet [16] inspired by the idea of convolution factorization, introduces asymmetric convolution block(ACB), which slightly improves performance and significantly reduces the computational cost. RepVGG [17] which is inspired by ResNet [35] applies RepVGG block with skip connections to replace the normal single convolution block. Later on, RepOptVGG [15] proposed to use the re-parameterized optimizer to replace the re-parameterized network architecture, which could even additionally dismiss the complexity in the training phase compared with RepVGG. In this study, the technique of re-parameterization shall be utilized to help SYENet to accelerate the inference.

3. Method

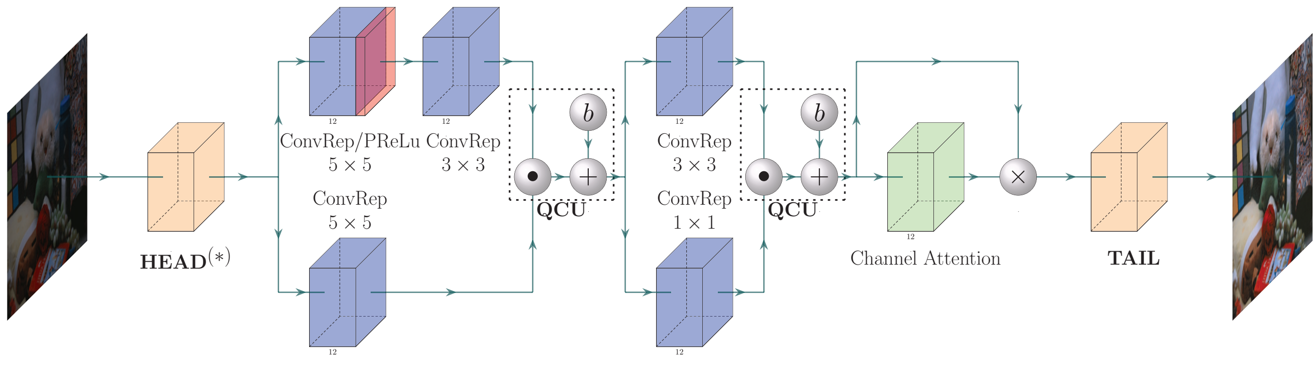
As the target platform for SYENet is mobile device, which has very limited hardware resources compared to cloud computing, each building block of SYENet should be carefully designed to reduce computation complexity while retaining the desired performance.

3.1. Texture generation and pattern selection

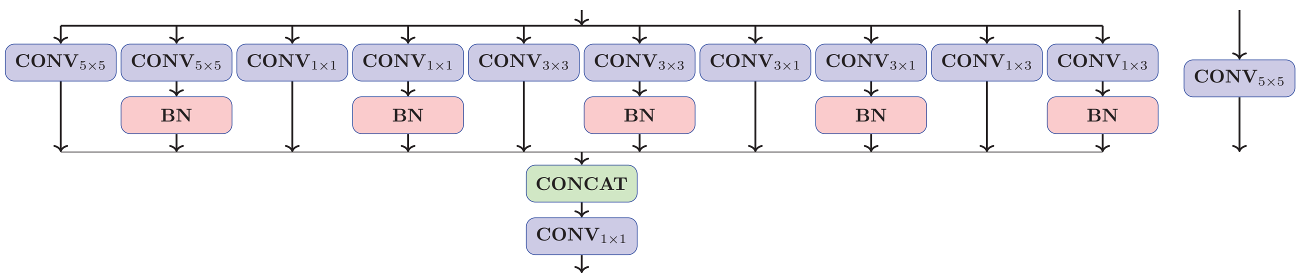
To reconstruct the desired images from the degraded input, texture and pixel pattern, which are compact representations and useful features, should be extracted and processed. The texture feature is the base for pixel prediction in SYENet. Pattern information reflected by color provides each pixel with classification information and is utilized to guide pixel prediction. Apparently, extracting the texture features as the regression task requires a deeper network for a larger receptive field than that of pattern information extraction as the classification task. Therefore, we use the asymmetric module with two branches for these tasks. The texture generation branch is designed to have two layers of convolutions, while the pattern selection branch only has one. For the same reason, the second asymmetric block is designed to have two branches with a 3×3 and 1×1 kernel convolution, respectively. The output of the two branches is shown in Fig. 3, and more examples can be found in Appendix K.

3.2. Quadratic Connection Unit (QCU): improving the capability of fitting arbitrary models

Typically, in the previous multi-branch networks, the fusion of outputs by different branches could be done by concatenation [2, 78] or element-wise addition followed by activation function [16, 24]. In this study, in order to effectively improve the representational power, a Quadratic Connec-



(a) Overall Architecture of SYENet: two $\odot$ operations are element-wise multiplications and $\otimes$ operation is channel-wise multiplication (* means some tasks may not require a head block to process). After the reparameterization, SYENet consists of 6 convolutions with only 5K parameters, excluding head and tail blocks.



(b) ConvRep block during training(left) and inference(right) phase, the training branches can be specifically designed for different requirements and applications.

Figure 2: Architecture of SYENet and the structure of ConvRep block in training (left) as well as inference (right) phase

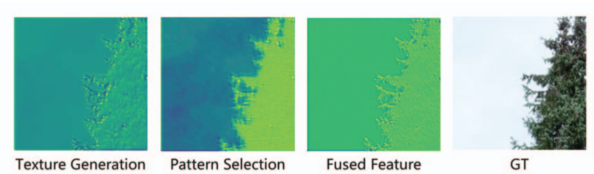


Figure 3: Complex texture feature with many details, simple pattern classification focusing on labeling and clustering pixels, fused results, and the ground truth

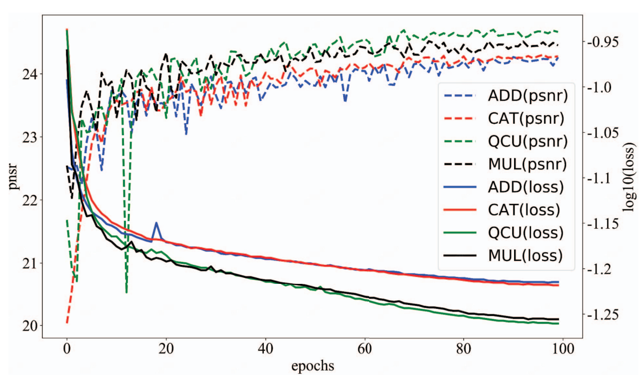


Figure 4: Faster convergence and higher PSNR by QCU compared with various fusion methods (addition(ADD), concatenation(CAT), multiplication(MUL)) in training SYENet for LLE task in LoL dataset. The QCU reaches higher PSNR and lower loss during the training.

tion Unit (**QCU**), as Eq. 1 where $\odot$ is an element-wise multiplication and $\oplus$ is element-wise addition, is employed for the fusion of the results by two branches F_1 and F_2. In big models with numerous channels, employing **QCU** may not make a difference because big models already have powerful expressiveness. However, for small models like SYENet this revision is rather vital.

$$\mathbf{QCU}(F_1, F_2) = (F_1 \odot F_2) \oplus \mathcal{B} \quad (1)$$

The formulation of F_1 and F_2 after re-parameterization shall be represented as linear form $KX + B$ due to convolution being linear transformation, so that the multiplied output should be in the quadratic form as $(\hat{K}X + \hat{B})(\tilde{K}X + \tilde{B})$.

In addition, NAFNet [9] revealed that activation could be replaced by multiplication in terms of providing nonlinearity towards the network.

However, we find that there exists the constraint or drawback of the above quadratic form by pure multiplication that the function must pass through two fixed position sets $(-\hat{B}/\hat{K}, 0)$ and $(-\tilde{B}/\tilde{K}, 0)$. Meanwhile, multiplication rather than addition could more easily enhance the influence of perturbations, which impairs robustness. To fix the two issues mentioned above, we add an element-wise learnable bias $\mathcal{B}$ to the fused output, which can impressively convert the expression to a more general form as $K_2X^2+K_1X+B$.

3.3. Outlier-Aware Loss: putting more focus on erroneously predicted pixels

In this study, applying the idea of Focal Loss [63] to regression problem, we propose a new loss function termed Outlier-Aware Loss $\mathcal{L}_{OA}$, as shown in Eq. 3, involving global information and putting more focus on the pixels that are badly predicted as the outliers. In Eq. 2, Δ is the difference between ground truth I^{GT} and the output by SYENet I^{SYE} in matrix form, and $\delta_{i,j}$ is the value of Δ in position (i, j). In Eq. 3, H and W are the output height and width. μ and σ^2, as the global information, are the mean and variance of Δ. b is the scale parameter defined by $2b^2 = \sigma^2$. α is a tunable hyperparameter assigned by the user. Compared with $\mathcal{L}_1$ loss, the loss in pixel (i, j) is multiplied by a weight $W_{i,j} = 1 - e^{-\alpha|\delta_{i,j}-\mu|^p/b}$. $W_{i,j}$ is proportional to $|\delta_{i,j} - \mu|$ and allows the model to focus on hard, erroneously predicted pixels. p is the norm number and is normally set to be 1 in low-level vision tasks implying the original loss to be optimized by W is $\mathcal{L}_1$ loss. Moreover, as shown in Table 4, Fig. 5, Fig. 6, and Fig. 7, Outlier-Aware Loss could improve the PSNR of the output images. A more detailed discussion of $\mathcal{L}_{OA}$ is in Appendix A.

$$\Delta = I^{SYE} - I^{GT} = \{\delta_{i,j} | i \in [0, H-1], j \in [0, W-1]\} \tag{2}$$

$$\mathcal{L}_{OA} = \frac{1}{HW} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} \left[|\delta_{i,j}|^p \times \left(1 - e^{-\alpha|\delta_{i,j}-\mu|^p/b}\right) \right] \tag{3}$$

3.4. Revised re-parameterization with enhancement by 1×1 convolution

All the convolution layers in SYENet shall be re-parameterized as Fig. 2b for inference. The convolution block in the training phase is expressed as Eq. 4.

$$I^{(out)} = \mathbf{CONV}_{1\times1}\Big(\mathbf{CAT}\big(\{\mathbf{CONV}_{\Phi}(I^{(in)}) | \Phi\}\big)\Big) \tag{4}$$

After the re-parameterization, the complex concatenation of several convolutions, half followed by batch normalization layers, shall be converted back to a single convolution layer as Eq. 5 for accelerating inference.

$$I^{(out)} = \mathbf{CONV}_{5\times5}(I^{(in)}) \tag{5}$$

Compared with the previous re-parameterization techniques, in SYENet, an improvement by one extra convolution layer with the kernel size of 1×1 is implemented after the concatenation to score the importance of each channel. Meanwhile, this structure can be re-parameterized like addition fusion. Compared with RepVGG block [17], our revised ConvRep block with 1×1 convolution, which simulates the function of channel attention, could improve the PSNR by 2.1932dB as shown in Table 4.

3.5. Simple Yet Effective (SYE) Network

The SYENet consists of 5 parts: head block, the first and second asymmetrical block, channel attention block, and tail block, which are assigned as $\mathbf{H}$, $\mathbf{A_1}$, $\mathbf{A_2}$, $\mathbf{CA}$ and $\mathbf{T}$. The head block is arranged for the preference of different tasks. The asymmetrical blocks are utilized to generate texture features and pattern information, which afterward shall be fused using multiplication. With the network input as $I^{(in)}$, the output of the first asymmetrical block $I^{(a_1)}$ and second $I^{(a_2)}$ are expressed as below, in which the subscript (c) and (s) represent the complex and the simple asymmetric branch respectively.

$$I^{(a_1)} = \mathbf{QCU}\Big(\mathbf{A}_1^{(c)}\big(\mathbf{H}(I^{(in)})\big), \mathbf{A}_1^{(s)}\big(\mathbf{H}(I^{(in)})\big)\Big) \tag{6}$$

$$I^{(a_2)} = \mathbf{QCU}\Big(\mathbf{A}_2^{(c)}(I^{(a_1)}), \mathbf{A}_2^{(s)}(I^{(a_1)})\Big) \tag{7}$$

The squeeze-and-excitation block is adopted and employed as the channel attention block, enhancing the expressiveness using global information to compensate for the disadvantage of the small receptive field. Hence the output of SYENet is expressed as Eq. 8, in which $\otimes$ is channel-wise multiplication.

$$I^{(out)} = \mathbf{T}\Big(\mathbf{CONV}\big(\mathbf{CA}(I^{(a_2)}) \otimes I^{(a_2)}\big)\Big) \tag{8}$$

4. Experiments

The experiments include sophisticated comparisons between SOTA methods with SYENet in (a)ISP, (b)SR, and (c)LLE issues and ablation studies. The evaluation metrics include PSNR and SSIM, but in order to assess the comprehensive performance of models considering both the image quality and efficiency, the comprehensive score Eq. 9 by

Method	Scale	#P	Avg latency(ms)	FPS(2K)	Set5	Set14	BSD100	BSD100 Score	Urban100	Urban100 Score
CISR [26]	×2	9.60M	1K+	<1	28.94/0.8160	26.78/0.7080	26.08/0.6590	-	24.93/0.7270	-
VSDR [52]	×2	0.65M	1K+	<1	37.53/0.9587	33.03/0.9124	31.90/0.8960	-	30.76/0.9140	-
DBPN [29]	×2	5.95M	1K+	<1	38.09/0.9600	33.85/0.9190	32.27/0.9000	-	32.55/0.9324	-
RDN [102]	×2	22.12M	1K+	<1	38.24/0.9614	34.01/0.9212	32.34/0.9017	-	32.89/0.9353	-
RCAN [101]	×2	12.47M	1K+	<1	38.27/0.9614	34.12/0.9216	32.41/0.9027	-	33.34/0.9384	-
HAN [72]	×2	64.61M	1K+	<1	38.27/0.9614	34.16/0.9217	32.41/0.9027	-	33.35/0.9385	-
DRLN [1]	×2	34.43M	1K+	<1	38.27/0.9616	34.28/0.9231	32.44/0.9028	-	33.37/0.9390	-
IPT [8]	×2	64.27M	1K+	<1	**38.37/-**	**34.43/-**	**32.48/-**	-	**33.76/-**	-
ESPCN [76](D0S3)	×2	0.191K	6.0	166	29.76/0.9190	28.96/0.8810	28.69/0.8650	1.737	26.38/0.8530	0.508
EDSR [62]	×2	1.37M	852.0	1	38.11/0.9601	33.92/0.9195	32.32/0.9013	1.874	32.93/0.9351	31.438
SRCNN [18]	×2	19.6K	168.0	5	36.66/0.9542	32.42/0.9063	31.36/0.8879	2.512	29.50/0.8946	1.373
eSR [71](C6D3S15)	×2	7.13K	119.0	8	36.58/0.9530	32.38/0.9050	31.25/0.8850	3.045	29.26/0.8910	1.389
SCSRN [42]	×2	50.0K	101.0	10	36.90/0.9565	32.59/0.9087	31.42/0.8904	4.541	29.63/0.8992	2.734
ABPN [20]	×2	33.5K	86.6	12	36.72/0.9556	32.49/0.9076	31.33/0.8891	4.675	29.39/0.8955	2.286
FSRCNN [19](D56S12M4)	×2	15.44K	87.6	11	36.74/0.9541	32.45/0.9070	31.34/0.8870	4.686	29.42/0.8950	2.356
HOPN [42]	×2	32.2K	61.7	16	36.27/0.9534	32.19/0.9049	31.11/0.8865	4.836	28.90/0.8885	1.627
TPSR-D2 [56]	×2	60.8K	105.0	10	37.18/0.9578	32.84/0.9112	31.64/0.8935	5.925	30.24/0.9073	6.126
FSRCNN [19](D32S6M4)	×2	5.78K	48.9	20	36.29/0.9510	32.20/0.9040	31.10/0.8840	6.018	28.91/0.8860	2.081
ESPCN [76](D64S32)	×2	24.48K	54.8	18	36.64/0.9530	32.46/0.9070	31.32/0.8870	7.286	29.37/0.8930	3.514
eSR [71](K3C1)	×2	0.105K	3.5	282	33.15/0.9280	30.16/0.8820	29.66/0.8620	11.422	26.94/0.8570	1.873
ESPCN [76](D22S32)	×2	9.2K	31.0	32	36.70/0.9530	32.47/0.9070	31.35/0.8870	13.426	29.44/0.8940	6.845
Compiler-Aware NAS [93]	×2	11K	31.6	27	37.19/0.9572	32.80/0.9099	31.60/0.8919	15.654	30.15/0.9054	15.100
FSRCNN [19](D6S3M1)	×2	1.08K	8.3	121	35.36/0.9430	31.52/0.8980	30.64/0.8780	18.740	28.01/0.8700	3.542
SYENet (**Ours**)	×2	4.932K	16.5	60	36.84/0.9564	32.62/0.9079	31.52/0.8907	**31.928**	30.37/0.9029	**46.681**
CISR [26]	×4	9.93M	1K+	<1	25.03/0.7020	23.88/0.5960	23.83/0.6590	-	21.86/0.5820	-
VSDR [52]	×4	0.65M	1K+	<1	31.35/0.8838	28.01/0.7674	27.29/0.7261	-	25.18/0.7524	-
RDN [102]	×4	22.27M	1K+	<1	32.47/0.8990	28.81/0.7871	27.72/0.7419	-	26.61/0.8028	-
RCAN [101]	×4	12.61M	1K+	<1	32.63/0.9002	28.87/0.7889	27.77/0.7436	-	26.82/0.8087	-
HAN [72]	×4	64.20M	1K+	<1	**32.64**/0.9002	28.90/0.7890	27.80/0.7442	-	26.85/0.8094	-
DBPN [29]	×4	10.43M	1K+	<1	32.47/0.8980	28.82/0.7860	27.72/0.7400	-	26.38/0.7946	-
IPT [8]	×4	64.41M	1K+	<1	**32.64/-**	**29.01/-**	27.82/-	-	**27.26/-**	-
DRLN [1]	×4	34.58M	1K+	<1	32.63/0.9002	28.94/0.7900	**27.83**/0.7444	-	26.98/0.8119	-
SRCNN [18]	×4	67.6K	167.0	5	30.48/0.8628	27.49/0.7503	26.90/0.7101	0.939	24.52/0.7221	0.990
EDSR [62]	×4	1.52M	418.0	2	32.46/0.8968	28.80/0.7876	27.71/0.7420	1.153	26.64/0.8033	7.475
eSR [71](C8D9S6)	×4	15.0K	131.0	7	30.62/0.8060	27.48/0.7510	26.93/0.7140	1.248	24.42/0.7180	1.099
ABPN [20]	×4	62K	50.1	20	30.61/0.8684	27.61/0.7578	26.94/0.7160	3.310	24.53/0.7275	3.347
SCSRN [42]	×4	73.9K	31.0	32	30.75/0.8719	27.75/0.7616	27.02/0.7188	5.955	24.69/0.7343	6.730
TPSR-D2 [56]	×4	61K	31.7	32	30.99/0.8761	27.85/0.7639	27.08/0.7211	6.349	24.81/0.7393	7.798
HOPN [42]	×4	41.3K	20.8	48	30.25/0.8598	27.35/0.7515	26.80/0.7115	6.564	24.27/0.7159	5.622
ESPCN [76](D64S32)	×4	27.3K	19.5	51	30.57/0.8580	27.50/0.7520	26.92/0.7150	8.268	24.42/0.7180	7.382
FSRCNN [19](D32S6M1)	×4	5.78K	11.8	84	30.16/0.8450	27.19/0.7420	26.74/0.7070	10.646	24.09/0.7020	7.21
ESPCN [76](D1S3)	×4	0.541K	5.65	176	28.93/0.8200	26.49/0.7250	26.25/0.6940	11.273	23.56/0.6800	7.803
FSRCNN [19](D44S12M4)	×4	13.26K	13.1	76	30.61/0.8610	27.52/0.7530	26.94/0.7160	12.654	24.44/0.7210	11.298
FSRCNN [19](D6S1M1)	×4	0.953K	5.7	125	29.31/0.8230	26.62/0.7300	26.41/0.6990	13.949	23.62/0.6830	8.331
eSR [71](K3C2)	×4	0.844K	3.68	271	28.64/0.8060	26.12/0.7120	26.13/0.6840	14.655	23.28/0.6680	8.011
ESPCN [76](D16S32)	×4	10.48K	11.3	88	30.59/0.8590	27.53/0.7530	26.95/0.7150	14.874	24.43/0.7190	12.918
SYENet (**Ours**)	×4	5.268K	9.92	100	30.33/0.8646	27.43/0.7532	27.02/0.7214	**18.670**	24.91/0.7299	**28.682**

Table 1: Comparison on super-resolution issue between the results by PSNR(dB), SSIM, and comprehensive score with SOTA: The methods are classified into big models with latency larger than 1K(ms) and small models. Big models are ranked by PSNR on BSD100 dataset, and small models are ranked by score in Eq. 9 on BSD100 dataset.

MAI Challenge [40] is introduced, in which constant C is employed for normalization.

$$\text{Score} = 2^{2\times\text{PSNR}}/(C \times \text{Latency}) \tag{9}$$

4.1. Implementation details

Training Setting. For **SR** task, the inputs are 128×128 patches with random augmentation of flips and rotations. The Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ and cosine annealing decay policy are utilized. Moreover, for **ISP** task, the input is preprocessed as Bayer Pattern with 256×256 resolution. Before the official training, an MAE-like [32] unsupervised warming-up phase is deployed to upgrade robustness as described in Appendix B. The **LLE** task follows the settings of the SR task except for the LoL [92] dataset.

Inference Setting. We use the Qualcomm Snapdragon 8 Gen 1 mobile SoC as our target runtime evaluation platform. The application we use to test the model runtime is AI benchmark [41, 44], which allows to load any custom TensorFlow Lite model [55] and run it on any Android device with all supported acceleration options. In our approach, we transform our Pytroch model into tflite model.

Datasets. The dataset for **ISP** task is MAI21 [43] adjusted using conversion by classical algorithm and warping by PDC-Net [81]. For the **SR** task, we use the DIV2K [80] for training and set5 [3], set14 [97], BSD100 [66], and Urban100 [37] for testing. For the **LLE** task, we use LoL [92].

4.2. Comparison with SOTA

In this study, we compare our proposed model with a variety of SOTA methods, from models with extreme com-

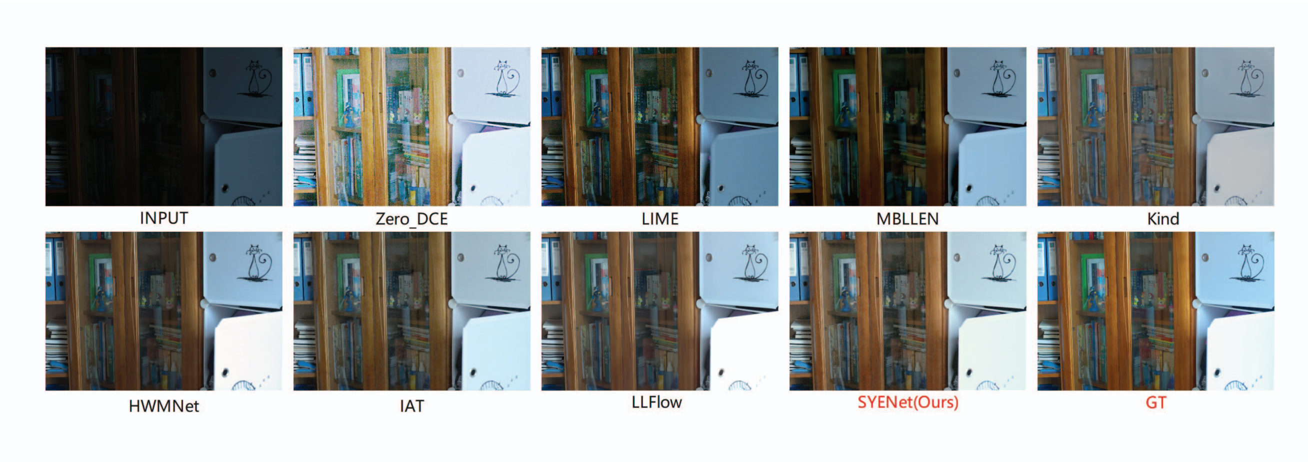


Figure 5: Low-light enhancement Comparison: The results reveal that our method could competitively recover the illuminance information. More comparisons of qualitative results are presented in the Appendix I.

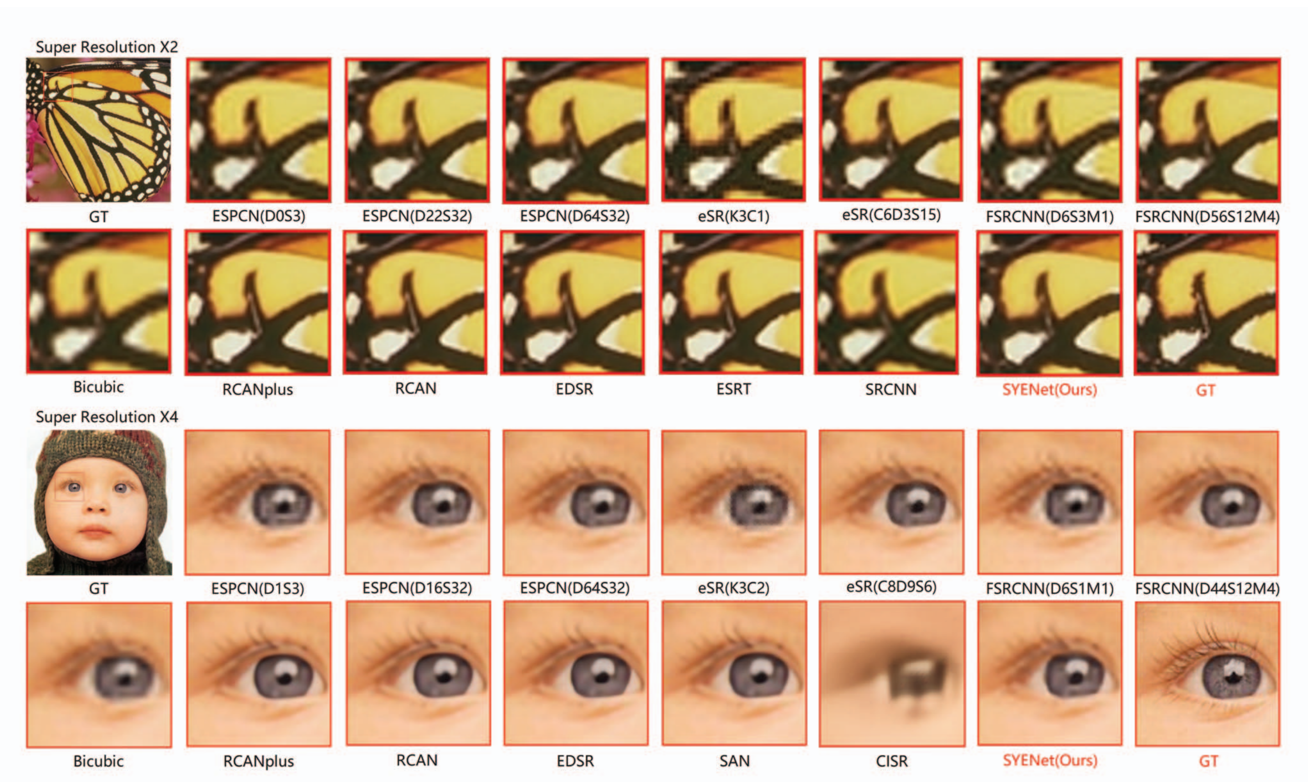


Figure 6: $\times 2$ and $\times 4$ SR comparisons with SOTA models: It is observed that our efficient model could generate output images with a similar quality compared with other large models. It is recommended to zoom in to observe the details.

plexity and distinct image quality to lightweight models with excellent efficiency and reasonably good output quality.

Super Resolution. As illustrated in Fig. 1b, and Table 1, SYENet achieves a competitive performance, which is roughly only 2dB lower than the highest PSNR but with only 0.17% of its parameters, as well as x100 times faster for inference. SYENet outperforms other lightweight models by 1 to 7dB, and as indicated by Table 1, SYENet gets far better scores than other lightweight models. The com-

Figure 7: Image signal processing comparisons with models from participators of MAI 2022 Challenge: Our model shows competitive performance compared with other efficient small networks, and the detailed quantitative comparisons are in Table 3. Even though the PSNR of our method is not the highest, the comprehensive performance measured by score (Eq. 9) is the highest. More comparisons of qualitative results are presented in the Appendix J.

Method	#P(M)	Mobile GPU latency(ms)	PSNR	SSIM
ZeroDCE [27]	0.08	858	14.83	0.531
UFormer [91]	5.29	-	16.27	0.771
3D-LUT [96]	0.60	-	16.35	0.585
Kind++ [100]	8.28	-	16.36	0.820
LIME [79]	-	-	16.76	0.650
RetiNexNet [92]	0.84	-	17.90	0.562
DRBN [95]	0.58	-	19.55	0.746
MBLLEN [22]	20.47	-	20.86	0.702
KIND [104]	8.16	-	21.30	0.790
Night Enhancement [48]	40.39	-	21.52	0.765
IPT [8]	115.63	-	22.67	0.504
IAT [12]	0.09	668	23.38	0.809
RCT [51]	-	-	23.43	0.788
MIRNet [28]	-	-	24.14	0.830
HWMNet [21]	66.56	-	24.14	**0.930**
MAXIM [82]	14.14	-	24.24	0.863
LLFlow [90]	17.42	-	**25.19**	0.850
SYENet (**Ours**)	**0.005**	33.4	22.59	0.807

Table 2: Comparison on low-light enhancement issue between the results by PSNR(dB) and SSIM with SOTA: The '-' mark in the Mobile GPU latency column refers that the latency of that model is larger than 1000ms.

Method	Model Size(MB)	PSNR	SSIM	GPU Runtime(ms)	Score
DANN-ISP	29.4	23.10	0.8648	583	0.13
MiAlgo	117	23.65	0.8673	1164	0.14
CASIA 1st	205	**24.09**	**0.8840**	1044	0.28
rainbow	1.0	21.66	0.8399	28	0.36
JMU-CVLab	0.041	23.22	0.8281	182	0.48
HITZST01	1.2	**24.09**	0.8667	482	0.60
ENERZAi	4.5	24.08	0.8778	212	1.35
MINCHO	0.067	23.65	0.8658	41.5	3.80
HITZST01	0.060	23.89	0.8666	34.3	6.41
ENERZAi	0.077	23.8	0.8652	18.9	10.27
MiAlgo	0.014	23.33	0.8516	**6.8**	14.87
SYENet(**Ours**)	0.029	23.96	0.8543	11.4	**21.24**

Table 3: Comparison on ISP performance by PSNR(dB) and SSIM with algorithms of MAI2022 ISP Challenge [45]: even though the PSNR of our method is not the highest, the comprehensive performance of our method measured by score (Eq. 9) is the highest.

parison between images by SYENet and other SOTA models with scale factors of $\times 2$ and $\times 4$ is shown in Fig. 6.

Low-light Enhancement. The enhanced low-light images obtained by a variety of models are shown in Fig. 5, and it is indicated that the images by SYENet could almost reach the objective quality of those by SOTA methods. More photos for comparison can be found in Appendix I. Finally, the objective measurements of SOTA algorithms and SYENet are shown in Table 2, which refers that SYENet achieves a competitive image quality at a rather faster speed using roughly only 0.01% of the size by the latest SOTA models.

Image Signal Processing. The comparison of performance and comprehensive scores by SYENet and the algorithms of MAI ISP Challenge participants is shown in Fig. 1a and Table 3. It is indicated that the comprehensive score by SYENet is significantly higher than the challenge-winning algorithm.

$\mathcal{L}_{OA}$	ConvRep	CA	QCU	Two-branch	PSNR	ΔPSNR
$\mathcal{L}_1$	√	√	√	√	24.7200	+0.1532
√	RepVGGBlock [17]	√	√	√	22.6797	+2.1932
√	×	√	√	√	24.6778	+0.1954
√	√	×	√	√	24.0936	+0.7796
√	√	√	ADD	√	24.5252	+0.3480
√	√	√	CAT+CONV	√	24.5427	+0.3305
√	√	√	MUL	√	24.7971	+0.0761
√	√	√	√	×	24.5510	+0.3222
√	√	√	√	√	24.8732	-

Table 4: Ablation study towards $\mathcal{L}_{OA}$(Outlier-Aware Loss) by $\mathcal{L}_1$(L1 loss), our re-parameterized convolution(ConvRep) by RepVGGBlock [17], CA(channel attention) by no CA, QCU(Quadratic Connection Unit) feature fusion by ADD(element-wise addition), MUL(element-wise multiplication), and CAT+CONV(concatenation followed by convolution) feature fusion, and two-branch asymmetric re-parameterized block by single branch re-parameterized block. The ablation study is conducted on ISP task.

Models	SYENet(ISP)			
Metric	PSNR ↑	LPIPS ↓	FID ↓	KID ↓
L1 Loss $\mathcal{L}_1$	24.7200	0.1681	28.0420	0.0095
Outlier-Aware Loss(Ours) $\mathcal{L}_{OA}$	**24.8732**	**0.1664**	**27.2182**	**0.0086**

Table 5: The performance of SYENet trained by two loss functions measured by different metrics: Outlier-Aware Loss improves PSNR as well as visual quality reflected by LPIPS, FID, and KID.

4.3. Ablation study

In the ablation study, the Outlier-Aware Loss $\mathcal{L}_{OA}$, ConvRep block as Fig. 2b, channel attention, QCU, and asymmetric branch block are degraded to be L1 loss $\mathcal{L}_1$, RepVGGBlock [16], no channel attention, three fusion methods (element-wise addition, concatenation plus convolution, and element-wise multiplication), and single branch block respectively. It shows that those components or methods indeed improve the PSNR. In addition, $\mathcal{L}_{OA}$ could improve the visual quality as Table 5.

5. Conclusion and Future Work

In this paper, we proposed SYENet, a novel and end-to-end mobile network for multiple low-level vision tasks with two asymmetric branches, **QCU**, revised re-parameter convolution, and channel attention. We also developed the **Outlier-Aware Loss** for better training. With these simple yet effective methods, SYENet is able to achieve 2K60FPS real-time performance on mobile devices for ISP, SR, and LLE tasks with the best visual quality.

While these initial results are promising, many challenges still remain. The most critical one is that the proposed network cannot handle all the low-level vision tasks, such as denoise and video SR. There's still room to improve the run-time efficiency by better utilization of limited hardware resources. In the future, we will focus on a more universal network architecture with reduced computation complexity.

References

[1] Saeed Anwar and Nick Barnes. Densely residual laplacian super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1192–1204, 2022.

[2] Mustafa Ayazoglu. Extremely lightweight quantization robust real-time single-image super resolution for mobile devices. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2472–2479, 2021.

[3] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012.

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[5] Yue Cao, Xiaohe Wu, Shuran Qi, Xiao Liu, Zhongqin Wu, and Wangmeng Zuo. Pseudo-isp: Learning pseudo in-camera signal processing pipeline from A color image denoiser. *CoRR*, abs/2103.10234, 2021.

[6] Kelvin C.K. Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Basicvsr: The search for essential components in video super-resolution and beyond. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2021.

[7] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3291–3300, 2018.

[8] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12299–12310, 2021.

[9] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. *arXiv preprint arXiv:2204.04676*, 2022.

[10] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

[11] Xiaojie Chu, Liangyu Chen, and Wenqing Yu. Nafssr: Stereo image super-resolution using nafnet. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1239–1248, June 2022.

[12] Ziteng Cui, Kunchang Li, Lin Gu, Shenghan Su, Peng Gao, Zhengkai Jiang, Yu Qiao, and Tatsuya Harada. You only need 90k parameters to adapt light: A light weight transformer for image enhancement and exposure correction, 2022.

[13] Linhui Dai, Xiaohong Liu, Chengqi Li, and Jun Chen. Awnet: Attentive wavelet network for image isp. *arXiv preprint arXiv:2008.09228*, 2020.

[14] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11065–11074, 2019.

[15] Xiaohan Ding, Honghao Chen, Xiangyu Zhang, Kaiqi Huang, Jungong Han, and Guiguang Ding. Re-parameterizing your optimizers rather than architectures. *arXiv preprint arXiv:2205.15242*, 2022.

[16] Xiaohan Ding, Yuchen Guo, Guiguang Ding, and Jungong Han. Acnet: Strengthening the kernel skeletons for powerful cnn via asymmetric convolution blocks. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[17] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13733–13742, 2021.

[18] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*, pages 184–199. Springer, 2014.

[19] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. *CoRR*, abs/1608.00367, 2016.

[20] Zongcai Du, Jie Liu, Jie Tang, and Gangshan Wu. Anchor-based plain net for mobile image super-resolution, 2021.

[21] Chi-Mao Fan, Tsung-Jung Liu, and Kuan-Hsien Liu. Half wavelet attention on m-net+ for low-light image enhancement. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 3878–3882, 2022.

[22] Jianhua Wu Feifan Lv, Feng Lu and Chongsoon Lim. Mbllen: Low-light image/video enhancement using cnns. *British Machine Vision Conference*, 2018.

[23] Qingxu Fu, Xiaoguang Di, and Yu Zhang. Learning an adaptive model for extreme low-light raw image processing. *arXiv preprint arXiv:2004.10447*, 2020.

[24] Michaël Gharbi, Jiawen Chen, Jonathan T. Barron, Samuel W. Hasinoff, and Frédo Durand. Deep bilateral learning for real-time image enhancement. *ACM Trans. Graph.*, 36(4), jul 2017.

[25] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

[26] Agus Gunawan and Sultan Rizky Hikmawan Madjid. Cisrnet: Compressed image super-resolution network. *arXiv preprint arXiv:2201.06045*, 2022.

[27] Chunle Guo, Chongyi Li, Jichang Guo, Chen Change Loy, Junhui Hou, Sam Kwong, and Runmin Cong. Zero-reference deep curve estimation for low-light image enhancement. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1777–1786, 2020.

[28] Hyewon Han, Soo-Whan Chung, and Hong-Goo Kang. MIRNet: Learning Multiple Identities Representations in Overlapped Speech. In *Proc. Interspeech 2020*, pages 4303–4307, 2020.

[29] Muhammad Haris, Greg Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1664–1673, 2018.

[30] Samuel Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan Barron, Florian Kainz, Jiawen Chen, and Marc Levoy. Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Transactions on Graphics*, 35:1–12, 11 2016.

[31] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *arXiv:2111.06377*, 2021.

[32] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022.

[33] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.

[34] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[35] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[36] Qiming Hu and Xiaojie Guo. Low-light image enhancement via breaking down the darkness. *CoRR*, abs/2111.15557, 2021.

[37] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5197–5206, 2015.

[38] Tao Huang, Lang Huang, Shan You, Fei Wang, Chen Qian, and Chang Xu. Lightvit: Towards light-weight convolution-free vision transformers. *arXiv preprint arXiv:2207.05557*, 2022.

[39] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *Proceedings of the 27th ACM International Conference on Multimedia (ACM MM)*, pages 2024–2032, 2019.

[40] Andrey Ignatov, Cheng-Ming Chiang, Hsien-Kai Kuo, Anastasia Sycheva, Radu Timofte, Min-Hung Chen, Man-Yu Lee, Yu-Syuan Xu, Yu Tseng, Shusong Xu, Jin Guo, Chao-Hung Chen, Ming-Chun Hsyu, Wen-Chia Tsai, Chao-Wei Chen, Grigory Malivenko, Minsu Kwon,

Myungje Lee, Jaeyoon Yoo, Changbeom Kang, Shinjo Wang, Zheng Shaolong, Hao Dejun, Xie Fen, Feng Zhuang, Yipeng Ma, Jingyang Peng, Tao Wang, Fenglong Song, Chih-Chung Hsu, Kwan-Lin Chen, Mei-Hsuang Wu, Vishal Chudasama, Kalpesh Prajapati, Heena Patel, Anjali Sarvaiya, Kishor Upla, Kiran Raja, Raghavendra Ramachandra, Christoph Busch, and Etienne de Stoutz. Learned smartphone isp on mobile npus with deep learning, mobile ai 2021 challenge: Report. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2503–2514, 2021.

[41] Andrey Ignatov, Radu Timofte, William Chou, Ke Wang, Max Wu, Tim Hartley, and Luc Van Gool. Ai benchmark: Running deep neural networks on android smartphones. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[42] Andrey Ignatov, Radu Timofte, Maurizio Denna, Abdel Younes, Ganzorig Gankhuyag, Jingang Huh, Myeong Kyun Kim, Kihwan Yoon, Hyeon-Cheol Moon, Seungho Lee, Yoonsik Choe, Jinwoo Jeong, Sungjei Kim, Maciej Smyl, Tomasz Latkowski, Pawel Kubik, Michal Sokolski, Yujie Ma, Jiahao Chao, Zhou Zhou, Hongfan Gao, Zhengfeng Yang, Zhenbing Zeng, Zhengyang Zhuge, Chenghua Li, Dan Zhu, Mengdi Sun, Ran Duan, Yan Gao, Lingshun Kong, Long Sun, Xiang Li, Xingdong Zhang, Jiawei Zhang, Yaqi Wu, Jinshan Pan, Gaocheng Yu, Jin Zhang, Feng Zhang, Zhe Ma, Hongbin Wang, Hojin Cho, Steve Kim, Huaen Li, Yanbo Ma, Ziwei Luo, Youwei Li, Lei Yu, Zhihong Wen, Qi Wu, Haoqiang Fan, Shuaicheng Liu, Lize Zhang, Zhikai Zong, Jeremy Kwon, Junxi Zhang, Mengyuan Li, Nianxiang Fu, Guanchen Ding, Han Zhu, Zhenzhong Chen, Gen Li, Yuanfan Zhang, Lei Sun, Dafeng Zhang, Neo Yang, Fitz Liu, Jerry Zhao, Mustafa Ayazoglu, Bahri Batuhan Bilecen, Shota Hirose, Kasidis Arunruangsirilert, Luo Ao, Ho Chun Leung, Andrew Wei, Jie Liu, Qiang Liu, Dahai Yu, Ao Li, Lei Luo, Ce Zhu, Seongmin Hong, Dongwon Park, Joonhee Lee, Byeong Hyun Lee, Seunggyu Lee, Se Young Chun, Ruiyuan He, Xuhao Jiang, Haihang Ruan, Xinjian Zhang, Jing Liu, Garas Gendy, Nabil Sabor, Jingchao Hou, and Guanghui He. Efficient and accurate quantized image super-resolution on mobile npus, mobile ai and aim 2022 challenge: Report, 2022.

[43] Andrey Ignatov, Radu Timofte, Maurizio Denna, Abdel Younes, Andrew Lek, Mustafa Ayazoglu, Jie Liu, Zongcai Du, Jiaming Guo, Xueyi Zhou, Hao Jia, Youliang Yan, Zexin Zhang, Yixin Chen, Yunbo Peng, Yue Lin, Xindong Zhang, Hui Zeng, Kun Zeng, Peirong Li, Zhihuang Liu, Shiqi Xue, and Shengpeng Wang. Real-time quantized image super-resolution on mobile npus, mobile ai 2021 challenge: Report. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2525–2534, 2021.

[44] Andrey Ignatov, Radu Timofte, Andrei Kulik, Seungsoo Yang, Ke Wang, Felix Baum, Max Wu, Lirong Xu, and Luc Van Gool. Ai benchmark: All about deep learning on smartphones in 2019. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3617–3635. IEEE, 2019.

[45] Andrey Ignatov, Radu Timofte, Shuai Liu, Chaoyu Feng, Furui Bai, Xiaotao Wang, Lei Lei, Ziyao Yi, Yan Xiang, Zibin Liu, Shaoqing Li, Keming Shi, Dehui Kong, Ke Xu, Minsu Kwon, Yaqi Wu, Jiesi Zheng, Zhihao Fan, Xun Wu, Feng Zhang, Albert No, Minhyeok Cho, Zewen Chen, Xiaze Zhang, Ran Li, Juan Wang, Zhiming Wang, Marcos V. Conde, Ui-Jin Choi, Georgy Perevozchikov, Egor Ershov, Zheng Hui, Mengchuan Dong, Xin Lou, Wei Zhou, Cong Pang, Haina Qin, and Mingxuan Cai. Learned Smartphone ISP on Mobile GPUs with Deep Learning, Mobile AI and AIM 2022 Challenge: Report. *arXiv e-prints*, page arXiv:2211.03885, Nov. 2022.

[46] Andrey Ignatov, Luc Van Gool, and Radu Timofte. Replacing mobile camera isp with a single deep learning model. *arXiv preprint arXiv:2002.05509*, 2020.

[47] Yani Ioannou, Duncan Robertson, Roberto Cipolla, and Antonio Criminisi. Deep roots: Improving cnn efficiency with hierarchical filter groups. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5977–5986, 2017.

[48] Yeying Jin, Wenhan Yang, and Robby T Tan. Unsupervised night image enhancement: When layer decomposition meets light-effects suppression. In *European Conference on Computer Vision*, pages 404–421. Springer, 2022.

[49] Eunah Jung, Nan Yang, and Daniel Cremers. Multi-frame gan: Image enhancement for stereo visual odometry in low light. In *Conference on Robot Learning*, pages 651–660, 2020.

[50] Zhang Kai, Danelljan Martin, Li Yawei, Timofte Radu, Liu Jie, Tang Jie, Wu Gangshan, Zhu Yu, He Xiangyu, Xu Wenjie, Li Chenghua, Leng Cong, Cheng Jian, Wu Guangyang, Wang Wenyi, Liu Xiaohong, Zhao Hengyuan, Kong Xiangtao, He Jingwen, Qiao Yu, Dong Chao, Luo Xiaotong, Chen Liang, Zhang Jiangtao, Suin Maitreya, Purohit Kuldeep, Rajagopalan A. N., Li Xiaochuan, Lang Zhiqiang, Nie Jiangtao, Wei Wei, Zhang Lei, Muqeet Abdul, Hwang Jiwon, Yang Subin, Kang JungHeum, Bae Sung-Ho, Kim Yongwoo, Qu Yanyun, Jeon Geun-Woo, Choi Jun-Ho, Kim Jun-Hyuk, Lee Jong-Seok, Marty Steven, Marty Eric, Xiong Dongliang, Chen Siang, Zha Lin, Jiang Jiande, Gao Xinbo, Lu Wen, Wang Haicheng, Bhaskara Vineeth, Levinshtein Alex, Tsogkas Stavros, Jepson Allan, Kong Xiangzhen, Zhao Tongtong, Zhao Shanshan, Hrishikesh P. S., Puthussery Densenand Jiji C. V., Nan Nan, Liu Shuai, Cai Jie, Meng Zibo, Ding Jiaming, Ho Chiu Man, Wang Xuehui, Yan Qiong, Zhao Yuzhi, Chen Long, Sun Long, Wang Wenhao, Liu Zhenbing, Lan Rushi, Umer Rao Muhammad, and Christian Micheloni. Aim 2020 challenge on efficient super-resolution: Methods and results. In Bartoli Adrien and Fusiello Andrea, editors, *Computer Vision – ECCV 2020 Workshops*, pages 5–40, Cham, 2020. Springer International Publishing.

[51] Hanul Kim, Su-Min Choi, Chang-Su Kim, and Yeong Jun Koh. Representative color transform for image enhancement. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4439–4448, 2021.

[52] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional

networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654, 2016.

[53] Dehui Kong, Ke Xu, Fang Zhu, Tongtong Zhu, Hengqi Liu, Yang Wei, and Jianjun Song. An efficient multi-scale method for single image super resolution. In *2020 International Wireless Communications and Mobile Computing (IWCMC)*, pages 1579–1583. IEEE, 2020.

[54] Yann LeCun. 1.1 deep learning hardware: Past, present, and future. In *2019 IEEE International Solid- State Circuits Conference - (ISSCC)*, pages 12–19, 2019.

[55] Juhyun Lee, Nikolay Chirkov, Ekaterina Ignasheva, Yury Pisarchyk, Mogan Shieh, Fabio Riccardi, Raman Sarokin, Andrei Kulik, and Matthias Grundmann. On-device neural net inference with mobile gpus. *arXiv preprint arXiv:1907.01989*, 2019.

[56] Royson Lee, L. Dudziak, M. Abdelfattah, Stylianos I. Venieris, H. Kim, Hongkai Wen, and N. Lane. Journey towards tiny perceptual super-resolution. *ECCV*, 2020.

[57] Royson Lee, Stylianos I. Venieris, Lukasz Dudziak, Sourav Bhattacharya, and Nicholas D. Lane. Mobisr: Efficient on-device super-resolution through heterogeneous mobile processors. In *The 25th Annual International Conference on Mobile Computing and Networking*, MobiCom '19, New York, NY, USA, 2019. Association for Computing Machinery.

[58] Mading Li, Jiaying Liu, Wenhan Yang, and Zongming Guo. Joint denoising and enhancement for low-light images via retinex model. In *International Forum on Digital TV and Wireless Multimedia Communications*, pages 91–99. Springer, 2017.

[59] Yanyu Li, Ju Hu, Yang Wen, Georgios Evangelidis, Kamyar Salahi, Yanzhi Wang, Sergey Tulyakov, and Jian Ren. Rethinking vision transformers for mobilenet size and speed. *arXiv preprint arXiv:2212.08059*, 2022.

[60] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1833–1844, 2021.

[61] Jinxiu Liang, Yong Xu, Yuhui Quan, Jingwen Wang, Haibin Ling, and Hui Ji. Deep bilateral retinex for low-light image enhancement. *arXiv preprint arXiv:2007.02018*, 2020.

[62] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017.

[63] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[64] Xin Liu, Yuang Li, Josh Fromm, Yuntao Wang, Ziheng Jiang, Alex Mariakakis, and Shwetak Patel. Splitsr: An end-to-end approach to super-resolution on mobile devices. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 5(1), mar 2021.

[65] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[66] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423. IEEE, 2001.

[67] Michaël Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *CoRR*, abs/1511.05440, 2015.

[68] Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 922–928. IEEE, 2015.

[69] Kangfu Mei, Juncheng Li, Jiajie Zhang, Haoyu Wu, Jie Li, and Rui Huang. Higher-resolution network for image demosaicing and enhancing. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3441–3448, 2019.

[70] Yiqun Mei, Yuchen Fan, and Yuqian Zhou. Image super-resolution with non-local sparse attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3517–3526, 2021.

[71] Pablo Navarrete Michelini, Yunhua Lu, and Xingqun Jiang. edge–sr: Super–resolution for the masses. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4019–4028, 2022.

[72] Ben Niu, Weilei Wen, Wenqi Ren, Xiangde Zhang, Lianping Yang, Shuzhen Wang, Kaihao Zhang, Xiaochun Cao, and Haifeng Shen. Single image super-resolution via a holistic attention network. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 191–207, Cham, 2020. Springer International Publishing.

[73] Siying Qian, Chenran Ning, and Yuepeng Hu. Mobilenetv3 for image classification. In *2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, pages 490–497, 2021.

[74] X. Ren, W. Yang, W. H. Cheng, and J. Liu. Lr3m: Robust low-light enhancement via low-rank regularized retinex model. *IEEE Transactions on Image Processing*, 29:5862–5876, 2020.

[75] E. Schwartz, R. Giryes, and A. M. Bronstein. Deepisp: Toward learning an end-to-end image processing pipeline. *IEEE Transactions on Image Processing*, 28(2):912–923, 2019.

[76] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016.

[77] Jinook Song, Yunkyo Cho, Jun-Seok Park, Jun-Woo Jang, Sehwan Lee, Joon-Ho Song, Jae-Gon Lee, and Inyup Kang.

7.1 an 11.5tops/w 1024-mac butterfly structure dual-core sparsity-aware neural processing unit in 8nm flagship mobile soc. In *2019 IEEE International Solid- State Circuits Conference - (ISSCC)*, pages 130–132, 2019.

[78] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015.

[79] Shiqiang Tang, Changli Li, and Xinxin Pan. A simple illumination map estimation based on retinex model for low-light image enhancement. In *2021 14th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–5, 2021.

[80] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 114–125, 2017.

[81] Prune Truong, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning accurate dense correspondences and when to trust them. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5714–5724, 2021.

[82] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxim: Multi-axis mlp for image processing. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5759–5770, 2022.

[83] Thang Vu, Cao Van Nguyen, Trung X Pham, Tung M Luu, and Chang D Yoo. Fast and efficient image quality enhancement via desubpixel convolutional neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[84] Muhammad Waleed Gondal, Bernhard Scholkopf, and Michael Hirsch. The unreasonable effectiveness of texture transfer for single image super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, September 2018.

[85] J. Wang, W. Tan, X. Niu, and B. Yan. Rdgan: Retinex decomposition based adversarial learning for low-light enhancement. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1186–1191, 2019.

[86] R. Wang, Q. Zhang, C. Fu, X. Shen, W. Zheng, and J. Jia. Underexposed photo enhancement using deep illumination estimation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6842–6850, 2019.

[87] W. Wang, C. Wei, W. Yang, and J. Liu. Gladnet: Low-light enhancement network with global awareness. In *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, pages 751–755, 2018.

[88] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[89] Yuzhi Wang, Haibin Huang, Qin Xu, Jiaming Liu, Yiqun Liu, and Jue Wang. Practical deep raw image denoising on mobile devices. In *European Conference on Computer Vision (ECCV)*, pages 1–16, 2020.

[90] Yufei Wang, Renjie Wan, Wenhan Yang, Haoliang Li, Lap-Pui Chau, and Alex Kot. Low-light image enhancement with normalizing flow. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2604–2612, 2022.

[91] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17662–17672, 2022.

[92] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. *arXiv preprint arXiv:1808.04560*, 2018.

[93] Yushu Wu, Yifan Gong, Pu Zhao, Yanyu Li, Zheng Zhan, Wei Niu, Hao Tang, Minghai Qin, Bin Ren, and Yanzhi Wang. Compiler-aware neural architecture search for on-mobile real-time super-resolution. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIX*, pages 92–111, 2022.

[94] Yunyang Xiong, Hanxiao Liu, Suyog Gupta, Berkin Akin, Gabriel Bender, Yongzhe Wang, Pieter-Jan Kindermans, Mingxing Tan, Vikas Singh, and Bo Chen. Mobiledets: Searching for object detection architectures for mobile accelerators. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3825–3834, 2021.

[95] Wenhan Yang, Shiqi Wang, Yuming Fang, Yue Wang, and Jiaying Liu. From fidelity to perceptual quality: A semi-supervised approach for low-light image enhancement. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3060–3069, 2020.

[96] Hui Zeng, Jianrui Cai, Lida Li, Zisheng Cao, and Lei Zhang. Learning image-adaptive 3d lookup tables for high performance photo enhancement in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(4):2058–2073, 2022.

[97] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010.

[98] Feng Zhang, Yuanjie Shao, Yishi Sun, Kai Zhu, Changxin Gao, and Nong Sang. Unsupervised low-light image enhancement via histogram equalization prior. *arXiv preprint arXiv:2112.01766*, 2021.

[99] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[100] Yonghua Zhang, Xiaojie Guo, Jiayi Ma, Wei Liu, and Jiawan Zhang. Beyond brightening low-light images. *Int. J. Comput. Vision*, 129(4):1013–1037, apr 2021.

[101] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 286–301, 2018.

[102] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2472–2481, 2018.

[103] Yonghua Zhang, Jiawan Zhang, and Xiaojie Guo. Kindling the darkness: A practical low-light image enhancer. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 1632–1640, 2019.

[104] Yonghua Zhang, Jiawan Zhang, and Xiaojie Guo. Kindling the darkness: A practical low-light image enhancer. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, page 1632–1640, New York, NY, USA, 2019. Association for Computing Machinery.

[105] Zhilu Zhang, Haolin Wang, Ming Liu, Ruohao Wang, Wangmeng Zuo, and Jiawei Zhang. Learning raw-to-srgb mappings with inaccurately aligned supervision. In *ICCV*, 2021.

[106] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9308–9316, 2019.