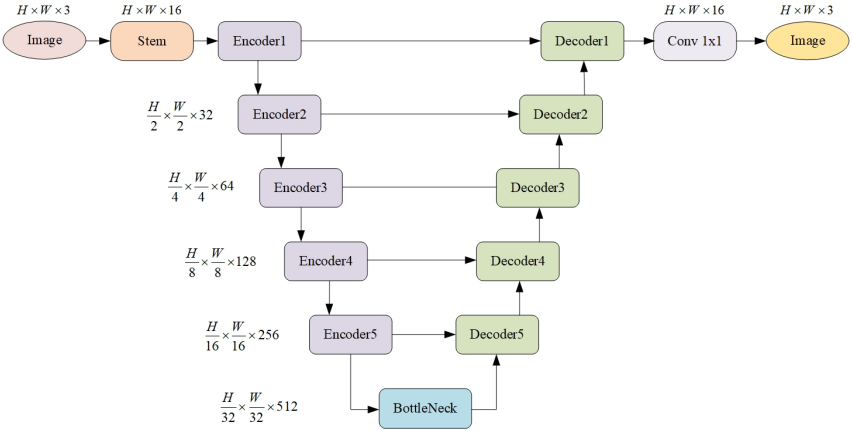
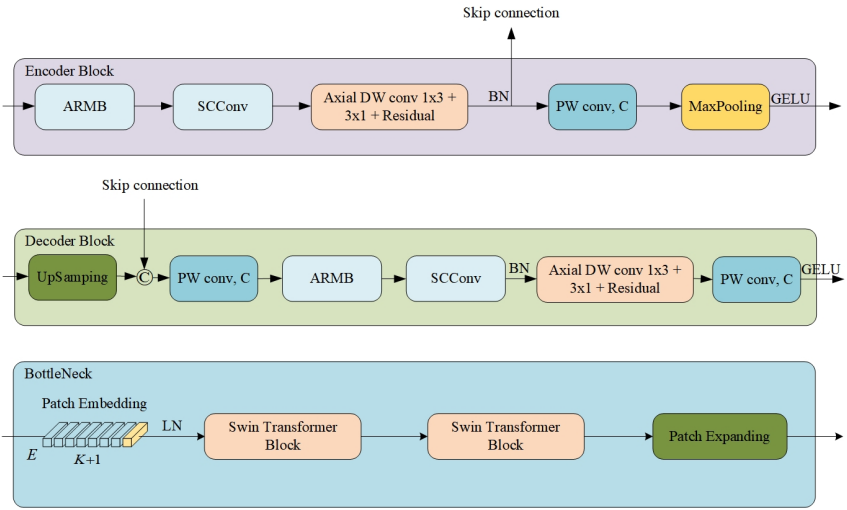
工作汇报

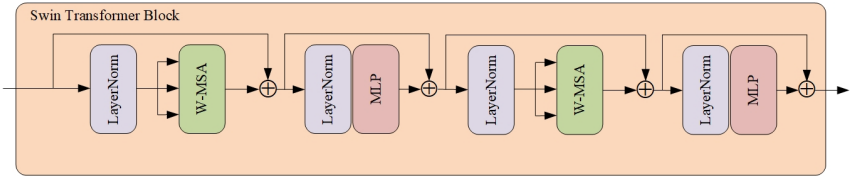
Gu Rui [1]

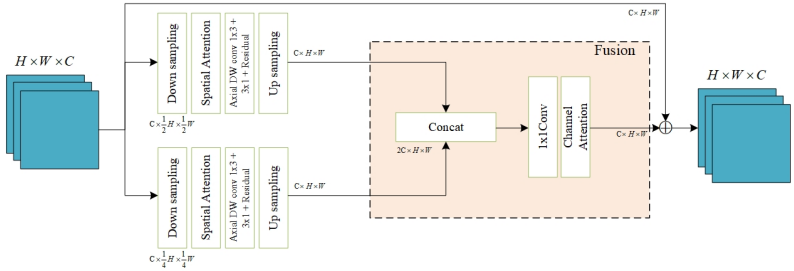
[1]School of Information Science and Engineering,
Lanzhou University
npukujui11@gmail.com

March 16, 2024


$H \times W \times 3$
Image
$H \times W \times 16$
Stem
Encoder1
Decoder1
$H \times W \times 16$
Conv 1x1
$H \times W \times 3$
Image
$\frac{H}{2} \times \frac{W}{2} \times 32$
Encoder2
Decoder2
$\frac{H}{4} \times \frac{W}{4} \times 64$
Encoder3
Decoder3
$\frac{H}{8} \times \frac{W}{8} \times 128$
Encoder4
Decoder4
$\frac{H}{16} \times \frac{W}{16} \times 256$
Encoder5
Decoder5
$\frac{H}{32} \times \frac{W}{32} \times 512$
BottleNeck


Skip connection
Encoder Block
ARMB
SCConv
Axial DW conv 1x3 + 3x1 + Residual
BN
PW conv, C
MaxPooling
GELU
Skip connection
Decoder Block
UpSamping
PW conv, C
ARMB
SCConv
BN
Axial DW conv 1x3 + 3x1 + Residual
PW conv, C
GELU
BottleNeck
Patch Embedding
E
K+1
LN
Swin Transformer Block
Swin Transformer Block
Patch Expanding


Swin Transformer Block
LayerNorm
W-MSA
LayerNorm
MLP
LayerNorm
W-MSA
LayerNorm
MLP


H × W × C
Down sampling
Spatial Attention
Axial DW conv 1x3 + 3x1 + Residual
Up sampling
C × H × W
C × 1/2 H × 1/2 W
Down sampling
Spatial Attention
Axial DW conv 1x3 + 3x1 + Residual
Up sampling
C × H × W
C × 1/4 H × 1/4 W
Fusion
Concat
2C × H × W
1x1Conv
Channel Attention
C × H × W
C × H × W
H × W × C

- **贡献**

✓ 首先，提出了一种结合卷积神经网络（CNN）和 Transformer 的并行架构用于弱光图像增强。

✓ 其次，提出了一个深度语义模块，该模块融合了 Swin Transformer 分支，使 CNN 分支能够有效捕获图像的长距离特征。

✓ 最后，将深度可分离卷积融合进 CNN 分支中，应用于轻量级网络用于提取图像的局部特征。

- 基于 PyTorch 进行模型的搭建、训练和评估;
- 基于 scikit-image 库计算 PSNR、SSIM 等评价指标;
- 构建 U-Net 基本架构模型;
- 实现 Swin Transformer 块中的LocalselfAttention类，PositionEncoding类，PositionEmbedding类

- **Low-light Dataset**

Name	Number	Format	Real/Syn	Video
LOL(?)	500	RGB	Real	
SCIE(?)	4,413	RGB	Real	
VE-LOL-L(?)	2,500	RGB	Real+Syn	

Table: Summary of paired training datasets. 'Syn' represents Synthetic.

- **Train**
  - ✓Baseline Model
  - ✓Ablation Study
- **Performance Evaluation**
  - ✓PSNR
  - ✓SSIM
  - ✓LPIPS
- **Loss Function**
  - ✓休伯损失函数和 SSIM 损失函数

$$L_{loss} = \alpha J_{Huber}(\delta) + \beta \mathcal{L}^{\text{SSIM}}(P) \tag{1}$$

  - ✓休伯损失函数，SSIM 损失函数，Perceptual 损失函数 (耗费更多训练时间)

$$L_{loss} = \alpha J_{Huber}(\delta) + \beta \mathcal{L}^{\text{SSIM}}(P) + \gamma \ell_{feat}^{\phi,j}(\hat{y}, y) \tag{2}$$

Thank you !