Micro-lens image stack upsampling for hyperspectral light fields

Paul Hasenbusch

October 19, 2025

Contents

1 Models

1.1 Prelimanries

1.1.1 Neural Networks

Definition 1 (Fully Connected Layer) *Let $A \in \mathbb{R}^{n\times m}$ and $b \in \mathbb{R}^m$. We call the mapping*

$$F : \mathbb{R}^n \to \mathbb{R}^m \ , \quad F(x) = Ax + b \ ,$$

a fully connected layer.

In order to refer to the architecture, thats is a fully connected layer with input dimension $n \in \mathbb{N}$ and output dimension $m \in \mathbb{N}$, whose weights are not fixed but subject to optimization, we write $F(n, m)$.

1.1.2 Convolutional Layers

Definition 2 (Convolution over multiple feature maps) *Let $X \in \mathbb{R}^{C\times n_1\times\ldots\times n_d}$ and $k \in \mathbb{R}^{C\times d_1\times\ldots\times d_d}$. The convolution of X with the kernel k, denoted by $k*X$ is given by*

$$(k * X)(p_0) = \sum_{i=1}^{c} \sum_{p\in R} k(c, p)X(c, p_0 + p) \ ,$$

where $R = \prod_{i=1}^{d} [0, d_i] \cap \mathbb{N}$, for all $p_0 \in \prod_{i=1}^{d} \mathbb{N} \cap [1, n_i - d_i]$.

Definition 3 (Convolutional Layer) *Let $X \in \mathbb{R}^{C\times n_1\times\ldots\times n_d}$, $k_1, ..., k_{C'} \in \mathbb{R}^{C\times d_1\times\ldots\times d_d}$ and $b \in \mathbb{R}^{C'}$. We call the mapping*

$$C : \mathbb{R}^{C\times n_1\times\ldots\times n_d} \to \mathbb{R}^{C'\times n_1\times\ldots\times n_d} \ , \quad C(X) = [k_1 * X, ..., k_{C'} * X] + b \ ,$$

a convolutional layer.

In order to refer to the architecture, thats is a convolutional layer with input channel dimension $C \in \mathbb{N}$ and output channel dimension $C' \in \mathbb{N}$, kernels $k_1, ..., k_{C'} \in \mathbb{R}^{C\times d_1\times\ldots\times d_d}$ and $b \in \mathbb{R}^{C'}$, which are not fixed but subject to optimization, we write

$$C(n, m, \text{kernel-size} = (d_1, ..., d_d), \text{padding} = p, \text{padding-mode} = m) \ .$$

Definition 4 (Residual Connection) *Let X be some set, $F = \{f : X \to X | f \text{ is a function}\}$ be the set of functions on X mapping back on X. The operation*

$$R : F \to F \ , \quad R(f)(x) = f(x) + x \ ,$$

for all $x \in X$, is called the residual mapping.

1.1.3 Transformers

Transformers operate on sequences of data $(x_k)_{k=1}^n$, where $x_k \in \mathbb{R}^d$. In the literature the elements of the input sequence are commonly referred to as tokens. Central to Transfomer models is the so-called attention mechanism. The tokens x_k are embedded into three different subspaces using linear mappings $Q, K, V \in \mathbb{R}^{d',d}$. The mappings Q and K are used to compute the attention scores among the members of the sequence, measuring the level of relevance of their respective information for each other

$$A_{ij} = \frac{\exp(x_i^T K^T Q x_j)}{\sum_{k=1}^n \exp(x_k^T K^T Q x_j)} \ . \tag{1}$$

The outputs are then computed for all $j = 1, ..., n$ by

$$y_j = \sum_{i=1}^n A_{ij} V x_i \ . \tag{2}$$

Note that by construction for all $j = 1, ..., n$ holds

$$\sum_{i=1}^n A_{ij} = 1 \ .$$

Definition 5 (Self-Attention) *Let $Q, K, V \in \mathbb{R}^{d',d}$. The operation described in equations (1), (2)*

$$SA : \mathbb{R}^d \times ... \times R^d \to R^d \times ... \times R^d \ , \quad SA(Q, K, V)(x_1, ..., x_k) = [y_1, ..., y_k] \ ,$$

is called self-attention.

To increase the expressiveness of Transformer models multiple self-attention mappings, called heads in the literature, are used in parallel to process the input sequence.

Definition 6 (Multi Headed Self-Attention) *Let $Q_h, K_h, V_h \in \mathbb{R}^{d',d}$ for $h = 1, ..., H$ and let $Q = (Q_1, ..., Q_H), K = (K_1, ..., K_H), V = (V_1, ..., V_H)$. The operation*

$$MSA(Q, K, V) = [SA(Q_1, K_1, V_1), ..., SA(Q_H, K_H, V_H)] \ ,$$

is called multi headed self-attention.

In order to refer to the architecture, thats is Multi Headed Self-Attention with dimension $d \in \mathbb{N}$ and number of heads $H \in \mathbb{N}$, whose weights are not fixed but subject to optimization, we write $\mathrm{MSA}(d, H)$.

Given a numer of heads $H \in \mathbb{N}$ generally the embedding dimension of each head is chosen as $\frac{d}{H}$.

Another key ingridient for Transformer models is layer normalization.

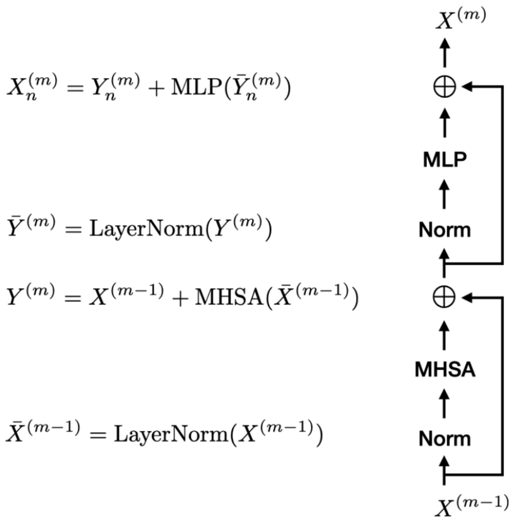
$X^{(m)}$
$X_n^{(m)} = Y_n^{(m)} + \mathrm{MLP}(\bar{Y}_n^{(m)})$
MLP
$\bar{Y}^{(m)} = \mathrm{LayerNorm}(Y^{(m)})$
Norm
$Y^{(m)} = X^{(m-1)} + \mathrm{MHSA}(\bar{X}^{(m-1)})$
MHSA
$\bar{X}^{(m-1)} = \mathrm{LayerNorm}(X^{(m-1)})$
Norm
$X^{(m-1)}$

Figure 1: Image taken from [**?**], Transformer Block architecture.

Definition 7 (Layer Normalization) *Let $\gamma \in \mathbb{R}$ and $\beta \in \mathbb{R}^d$. The operation given by*

$$LN \colon \mathbb{R}^d \to \mathbb{R}^d \ , \quad LN(x) = \gamma \bar{x} + \beta \ \textit{where} \ \bar{x}_{ki} = \frac{1}{\sqrt{var(x_k)}}(x_{ki} - \sum_{j=1}^{d} x_{kj})$$

is caled layer normalization.

Typically Multi Headed Self-Attention is used in transfomer blocks, the architecture is outlined in figure 1.

First layer normalization is applied to the inputs before the Multi Headed Self-Attention is being performed. The input is then added back to the outputs via a residual connection. The intermediate throughputs then undergo a second round of layer normalization, the tokens are then processed individually by a neural network. Mathematically this can be summarized by

$$\text{TransfomerBlock} = R\big(\Phi \circ \text{LN}\big) \circ R\big(\text{MSA}(d, H) \circ \text{LN}\big)$$

where $\Phi : \mathbb{R}^d \to \mathbb{R}^d$ is some neural network, typically with only a few number of layers.

1.2 General Model Architecture

Independent of the domain of application, a general architectural choice that can be observed in all super resolution models, is that the architecture is made up of three components. A shallow feature extraction module H_S, a deep feature extraction module H_D and an image reconstruction module H_{IR}. Typically the model architecture is conceptualized as follows

$$H = H_{IR} \circ R(H_D) \circ H_S \ . \tag{3}$$

The shallow feature extraction module H_S scales the channel dimension of the input to a higher dimension, which is used throughout the majority of the network. Additionally it extracts low frequency features. The module is usually composed of only one or few convolutional layers.
The deep feature extraction module H_D forms the main part of the model. It is supposed to recover high frequency information. Here is where different architectures proposed in the literature vary the most, convolutional layers, transformer models and various combinations thereof have been tried out.
Note the residual connection in equation (3), the rational behind this being that this way the low frequency features extracted by H_S can bypass the deep feature extraction module H_D. The image reconstruction module H_{IR} maps the input back to the original channel dimension and scales the spatial dimension to the desired size. It has been experimentally confirmed that better results are achieved when scaling is done at the end, rather than processing the already spatially upsampled input. To this end usually transposed convolutional layers or pixel sshuflling layers are employed.

1.3 Single Image Super Resolution

1.3.1 Deep Residual Channel Attention Network

The Deep Residual Channel Attention Network (DRCAN) proposed by Zhang et al. [?], the channel attention mechanism is introduced to single image super resolution. Channel Attention enables the network to dynamically assess which feature maps / channels are more important or need more refinement. This is achieved by processing the globally pooled average of the feature maps using a lightweight network and then reweighing the feature maps based thereon.

The overall model architecture is depicted in figure 2. The input image X is first processed via an initial convolutional layer

$$F_0 = C(3, 64, \text{kernel-size} = 3, \text{padding} = 1)(X) \ .$$

The following convolutional layers used in the architecture of the DRCAN are of the form

$$C = C(64, 64, \text{kernel-size} = 3, \text{padding} = 1, \text{padding-mode} = \text{zero}) \ ,$$

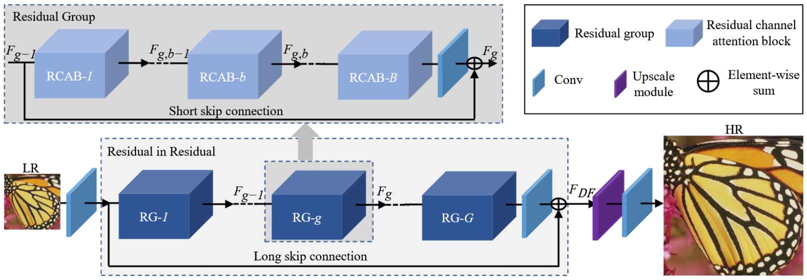

Figure 2: Image taken from [**?**], DRCAN model architecture.

that is 64 ingoing feature maps processed by 64 quadratic kernels of size 3 × 3 with zero-padding of size 1, so that feature map sizes are conserved throughout the model. The inial features F_0 are then further processed by a network with a residual in residual architecture

$$F_1 = H_{RIR}(F_0) \ .$$

For low-frequency features to bypass the deep feature extraction, a residual connection is used before the upsampling is performed

$$F_2 = F_0 + F_1 \ .$$

The final features F_2 are then upsampled using transposed convolutional layers.
The H_{RIR} network is composed of 10 Residual Groups followed by a final convolutional layer, that is

$$H_{RIR} = C \circ H_{RG} \circ ... \circ H_{RG} \ .$$

The Residual Groups (RG) are again composed of 20 Residual Channel Attention Blocks followed as well by a convolutional layer, the structure is encapsuled in a residual connection

$$H_{RG} = R\big(C \circ H_{RCAB} \circ ... \circ H_{RCAB}\big) \ .$$

The Residual Channel Attention Block (RCAB) depicted in figure 3, is made up of two convolutional layer, with a ReLU activation function in between, followed by a channel attention module, the output is then added back to the input again via a residual connection

$$H_{RCAB} = R\big(H_{CA} \circ C \circ \mathrm{ReLU} \circ C\big) \ .$$

The channel attention mechanism depicted in 4. The information of a feature map is first condesated into a single value by using global pooling

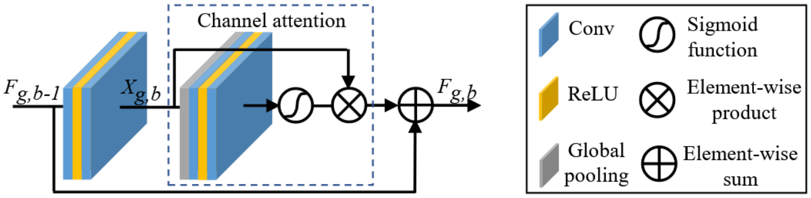

Figure 3: Image taken from [?], architecture of RCAB module.

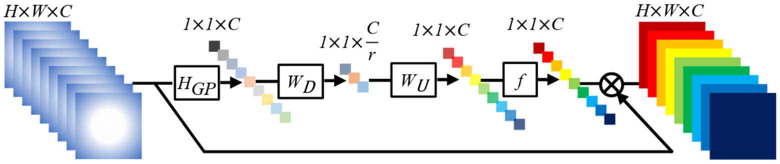

Figure 4: Image taken from [?], Channel Attention machanism.

$$z_c = H_{GP}(x_c) = \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} x_c(i, j) \ ,$$

with the input $X = [x_1, ..., x_C] \in \mathbb{R}^{C \times H \times W}$. The vector $z \in \mathbb{R}^C$ is then processed by a two-layer neural network

$$\Phi = \sigma \circ F(4, 64) \circ \text{ReLU} \circ F(64, 4) \ ,$$

the sigmoid activation function is applied at last, in order to squash the attention scores into the interval $[0, 1]$. Channel attention the weights the inputs according to the attention scores

$$H_{CA}(X) = \Phi \circ H_{GP}(X) \cdot X \ .$$

1.3.2 Shifted Window Transfomer Image Restoration

The SWinIR model proposed by Liang et al. [?], makes use of the shifted window transfomer architecture introduced by Liu et al. [?]. While the model does not employ the hierarchical structure of the original architecture, it makes extensive use of the shifted window mechanism. The model architecture is depicted in figure **??**.

1.3.3 Hybrid Attention Transfomer

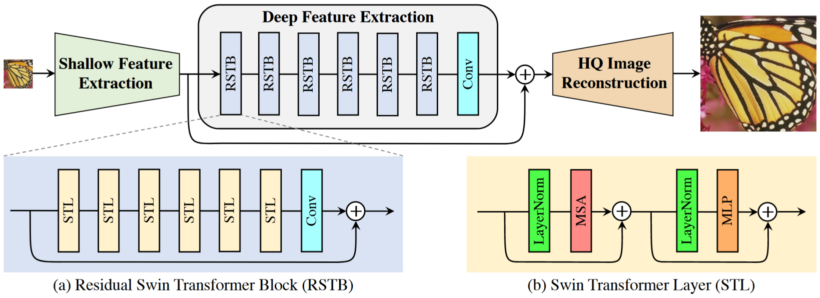

Figure 5: Image taken from [**?**], architecture of SWinIR model.

2 Training

2.1 Preprocessing the Data

2.2 Training Methods

2.2.1 Single Image Super Resolution

2.2.2 Light Field-, Hyperspectral Image- and Spectral Super Resolution Methods

2.2.3 Diffusion Models