隐式神经的理解和总结

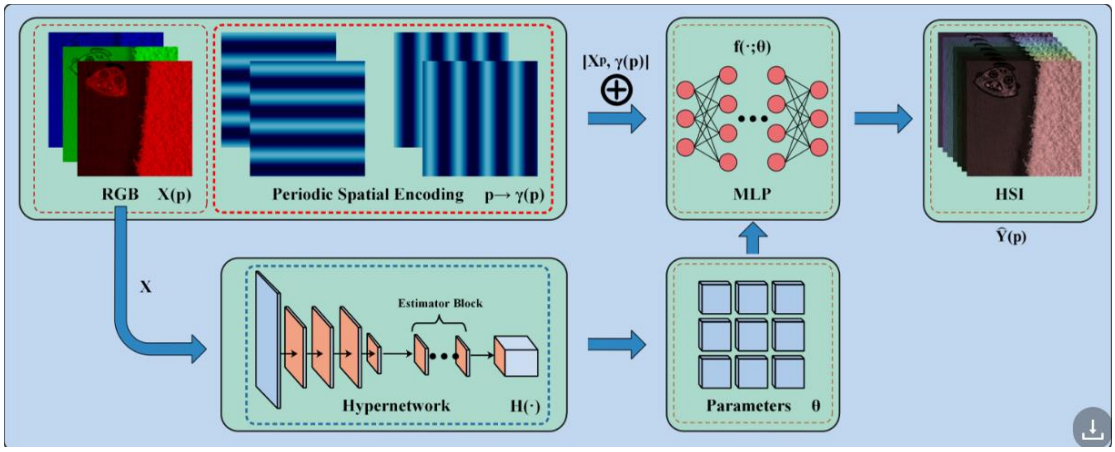
1.个人理解：
隐式神经简而言之，就是将离散的信号表示成连续的函数。在图像处理方面，实现图像的超分辨提供了一个有利的工具，在一定程度上可以将图像无限上采样，故可以实现无限大的分辨率。但隐式神经也同样存在一定的缺点，如由于在计算某个位置的像素值时采用的线性计算的方法，容易使图像变得过于平滑，以及计算量大，耗时等等

2.图像处理中简要的步骤:
对拿到的 LR 图像进行编码，拿到一个特征图，然后给定任意一个坐标，利用 MLP 预测高分辨率的像素值，实现图像的超分。
接下来，我将主要通过举一些文章中的应用事例，体会隐式神经，INR 在深度学习中的应用优势和特点。
（MLP：多层感知机中隐藏层多由 RELU 为激活函数，但通过查找资料，发现这样处理容易造成图像过于平滑，细节丢失严重，而相应使用正弦函数则结果会有提升。）

3.我的调研体会：
（1）Implicit Neural Representation Learning for Hyperspectral Image Super-Resolution（IEEE）

任务：高光谱的重建往往分为两种，一是融合，二是单个图像的超分，前者将高空间分辨率的 MSI 和空间分辨率较低的 HSI 合并，生成高空间分辨率的 HSI，而后者旨在从其对应的 RGB 图像重建 HIS，本篇文章即为后面一种。

模型结构：将 RGB 图像作为 MLP 的输入，学习里面的权重参数，其中加入了图像的位置编码，即用三角函数表示每个像素点 p 编码为向量 γ（p），使得目标像素值的相邻像素都可以由所有隐藏层中的神经元所访问，更好复原高频信息。



笔者认为，本篇文章的核心在于下方 H（.）和 θ 的计算，使得隐式神经网络能够将 RGB 图像复原为 HIS，接下来将详细讲解
首先 RGB 先作用一个跨步的卷积层，减小规模，之后作用于估计块（estimator block）保

证输出的特征图和输入大小一致，最后得到 MLP 的权重和偏置。可以看到最后的参数取决于输入本身。
之后是 MLP 参数 θ 和上步输出的网格之间存在的一些关系。
最终输出的参数网格是小于原 RGB S 倍，于是将参数网格分成很多 S*S 的网格，例如原 RGB 网络为 64*64，S=4 时，输出参数网格是 16*16，再划分成 16 个 4*4 的网络，最后得到的参数 θ 网络维度为【B，num-params，16*16，64】，其中 B 代表 batch，num-params 为 MLP 所有的参数数量。由于文章并未明确给出 MLP 中参数和计算后的网格之间一一对应的关系，因此这里在实现像素值预测时参数的选择还尚不明晰，笔者推测应该需要追踪代码去进一步研究。

损失：L1=|Y-M（X）|,Y 为 HIS 图像，M（X）为 RGB 图像恢复到 HIS 的图像

疑问：知道这一步就是为了求得自适应的隐式神经里的参数 θ，但不理解最后网络维度为什么是【B，num-params，16*16，64】，这里跟之前划分参数网格有什么直接联系还没能解决。

文章知识补充：（"estimator block"（估计器模块）通常是指一个模型或子模块，它用于估计某个特定的量或变量。具体来说，"estimator block"的功能和用途可能因应用场景而有所不同，但一般包括以下几个方面：

1. 损失函数估计：在训练过程中，估计损失函数的值，帮助优化器调整模型参数以最小化损失。
2. 输出预测：在推理阶段，生成模型的最终预测结果。对于分类任务，这可能是分类概率；对于回归任务，这可能是连续值。
3. 中间表示估计：在多阶段模型中，估计中间特征或表示，用于后续处理。例如，在图像处理任务中，估计器模块可能会生成中间的特征图，供后续的卷积层使用。
4. 不确定性估计：在一些应用中，估计器模块还可能用于估计模型预测的不确定性，帮助判断预测的可靠性。

具体实现上，估计器模块通常包含一组神经网络层，这些层根据输入数据生成相应的输出。这些层可以是卷积层、全连接层、归一化层等，视具体任务和网络架构而定。）

空间编码：设计的空间编码可以显著提高恢复细粒度细节的性能

$$\gamma_k(p) = \left[\cos\left(2^k\pi x\right), \sin\left(2^k\pi x\right), \cos\left(2^k\pi y\right), \sin\left(2^k\pi y\right)\right]. \tag{7}$$

Afterward, we set $k = 0, \ldots, N-1$ to encode kth frequency component of the 2-D pixel position $p = (x, y)$ as a vector of sinusoids with different frequencies

$$\gamma(p) = \left[\gamma_0(p), \ldots, \gamma_{N-1}(p)\right]. \tag{8}$$

具体操作如图所示

不适定性问题（Ill-posed problem）通常包含三个方面：

1. 解的存在性：是否存在一个解满足给定的条件。
2. 解的唯一性：是否该解是唯一的。
3. 解的稳定性：在输入数据发生微小变化时，解是否会发生巨大的变化。

在图像超分辨率任务中，这三个方面的不适定性主要体现在以下几点：

1. 解的多样性：给定一个低分辨率图像，可以有多个高分辨率图像作为其对应的高分辨率版本。例如，模糊的低分辨率图像可能对应多个细节不同的高分辨率图像。这说明解的唯一性得不到保证。

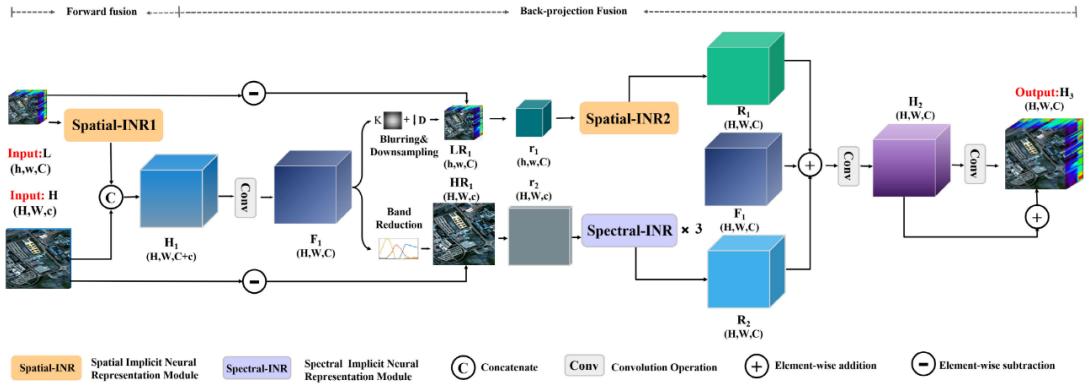
2. 信息的丢失：在从高分辨率图像降采样到低分辨率图像的过程中，往往会丢失一些细节信息。这些丢失的信息在重建过程中很难准确地恢复回来，导致解的存在性和稳定性问题。
3. 噪声影响：低分辨率图像中可能存在噪声，这些噪声在重建过程中会被放大，影响超分结果的质量。这意味着解的稳定性不强。

总结：这篇文章架构不复杂，主要在于隐式神经 MLP 中参数的计算，其中位置编码在恢复高频细节具有重要作用，接下来参数网络的计算需要近一步的探讨学习。

（2）SS-INR(IEEE)

任务：解决超分的不适定问题，获得质量更高的 HIS 或 MSI，受 INR 启发分别提出空间和光谱上的重建模型。和（1）对比此任务属于融合范畴。

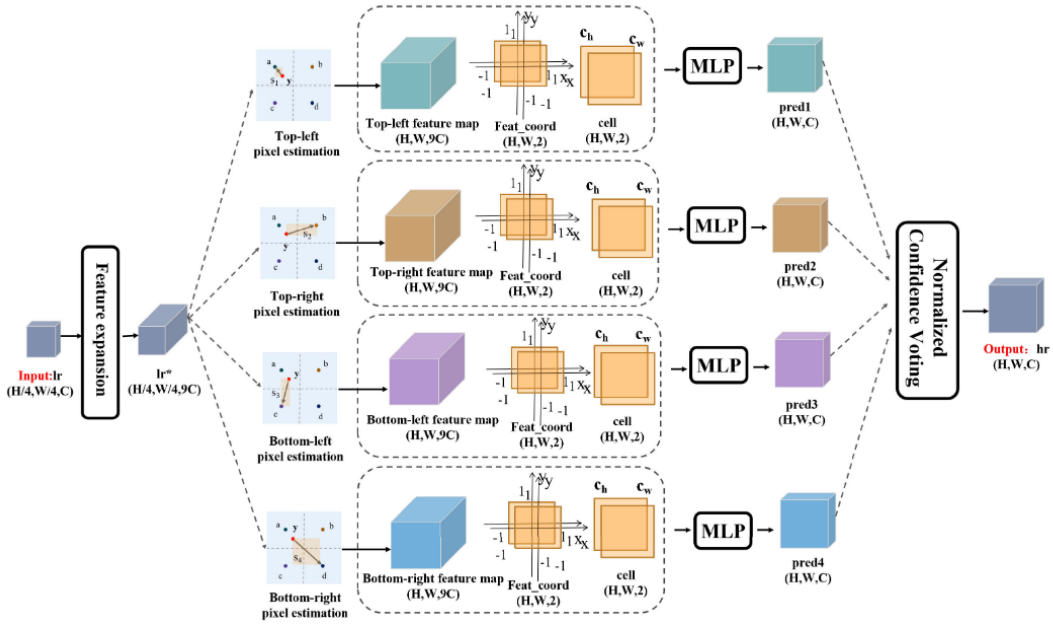
模型结构：SS-INR 包含两个过程，一个是前向的融合过程，输入 HSI 首先使用空间 inr 进行空间上采样，以克服空间分辨率差异，同时与 MSI 进行初始融合；另一个是反向注入的融合过程，探索了空间和光谱退化过程，并将它们用作纠错的先验知识。



g. 2. Overall structure diagram of the proposed SS-INR. The main building blocks of the proposed SS-INR architecture are two parts: FF and BPF.

总体结构为从左到右如上图所示，不过多赘述，下面主要详细分析出现的三处 INR 网络

1.



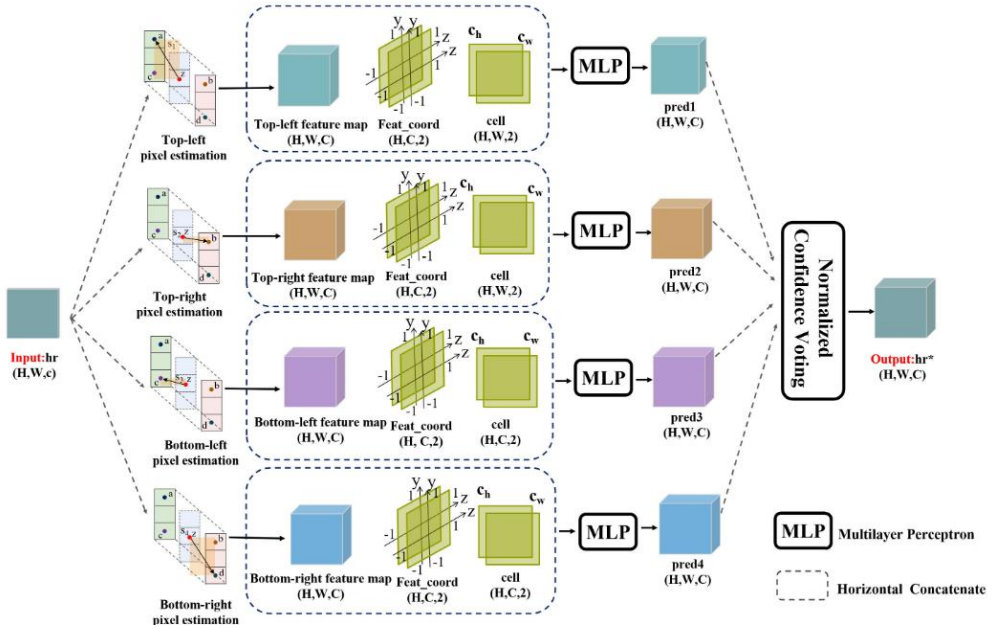
这里首先将输入做一个通道特征的延拓，后在四个方向分别计算预测点的像素值。

预测 y 点像素值由周围四点决定，即

$$X(y)=\sum_{m\in\{00,01,10,11\}}\frac{s_m}{s}f_\theta\left(a_m,y-a_m^{\text{coord}}\right)\qquad(5)$$

sm 分别为 y 与 abcd 四点形成矩形面积

2.



类似的，光谱上的隐式神经表示为在通道方向对某波段像素值进行预测，为了防止随位置移

$$Y(x)=\sum_{m\in\{00,01,10,11\}}\frac{s_m}{s}g_\theta\left(b_m,x-b_m^{\text{coord}}\right)\qquad(12)$$

动变化剧烈，采用周边四个方向的像素值加权预测

知识补充：上面两张模型图发现都存在命名为 cell 的图形，仔细阅读文章会发现和像素的

宽和高有关，这也是笔者在阅读时始终存在的疑问，为什么一个正正方方的像素值还要讨论它的长和高，后来通过查找资料，外加自己对文章的理解如下

传统 INR 方法存在局限性

1. 像素中心坐标方法：在传统的图像重建方法中，每个像素值通常是基于其中心坐标来计算的。这种方法的一个主要局限是它忽略了像素实际覆盖的区域，只关注单个点（像素中心），从而丢失了像素边界和面积的信息。

空间-INR (Spatial Implicit Neural Representation)

Spatial-INR 使用了一种称为 cell decoding 的策略，这种策略通过引入像素块的高度和宽度信息来改进图像重建过程。

具体步骤

1. 像素块（Cell）：在图像重建任务中，每个像素实际上覆盖了一个小区域（即一个像素块）。例如，在一个 100x100 分辨率的图像中，每个像素块可能覆盖 1/100 的图像宽度和 1/100 的图像高度。
2. 高度和宽度：每个像素块不仅仅是一个中心点，还包含了该块的物理尺寸信息，即高度和宽度。这些尺寸信息提供了像素覆盖的实际区域。
3. 将尺寸信息引入网络：在 Spatial-INR 方法中，这些高度和宽度信息作为辅助输入，和像素的中心坐标一起被输入到网络中。这种做法允许网络了解每个像素块的实际大小和形状，从而在重建图像时能够更好地保留细节和边界信息。

举例说明

假设你有一个 3x3 的小图像，传统方法可能仅仅使用像素中心（如 (0.5, 0.5), (1.5, 0.5), ...）来确定像素值。而使用 cell decoding 方法，你不仅使用这些中心坐标，还将每个像素块的大小（如高度和宽度均为 1）引入网络。这样，网络在重建时可以更准确地表示图像中的每个像素块，避免信息丢失。

这样就更加准确地恢复 HIS 或 MSI 图像

总结：本篇文章主要通过在空间和光谱域上对图像进行优化，INR 方面的特点主要是在计算中心坐标关注到了周围四点坐标的像素值，并将 space spectral 两个方向分别计算。这里我觉得可以多多关注提到的反向过程，这里为什么存在这一步骤可以总结如下

1.模拟退化过程：帮助模型理解信息在退化过程中的丢失。

2.计算残差：提供重要的先验知识，揭示退化过程中丢失的细节和信息。

（3）Two Spectral－Spatial Implicit Neural Representations for Arbitrary-Resolution Hyperspectral Pansharpening（IEEE）

任务:提出了两种用于任意分辨率的泛锐化的光谱空间 INR:一种是朴素光谱空间泛锐化 INR (NaivePINR)，另一种是动态光谱空间泛锐化 INR (DynamicPINR)。

模型结构：

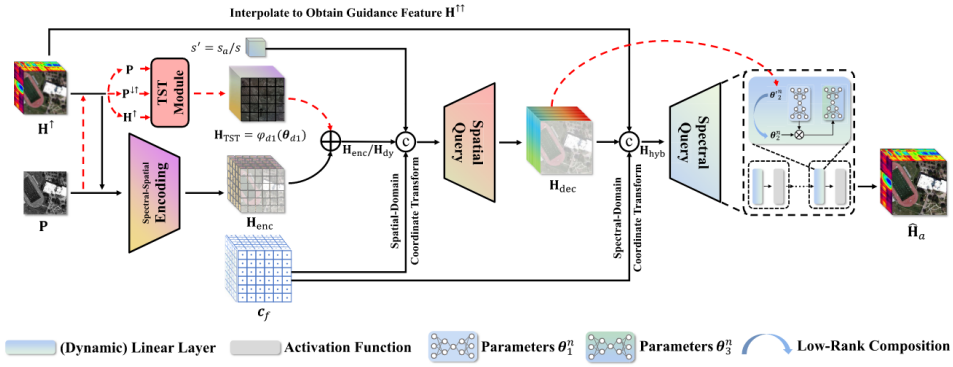


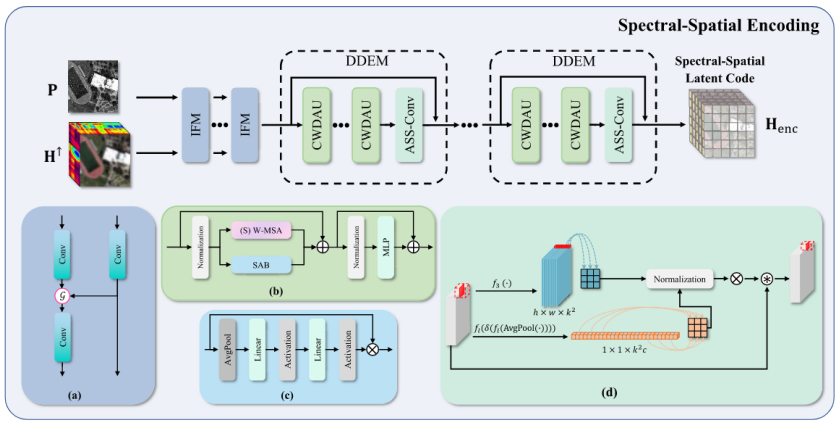
Fig. 2. Overall architectures of our spectral–spatial pansharpening INRs for ARHS pansharpening, where red dashed lines and connections are used for our DynamicPINR. ⊕ denotes the element-wise sum, ⊗ denotes the element-wise product, and © denotes the concatenation operation.

其中红色虚线为 DPINR，是在 NPINR 基础上做的近一步改善。

NPINR

Spectral－Spatial Encoding

接下来主要分析空间光谱编码这一操作的实现。



Architecture of spectral–spatial encoding. ⊛ denotes the dynamic convolution. (a) IFM. (b) CWDAU. (c) SAB. (d) ASS-Conv.

首先 IFM 用两个不同的卷积层提取 P,H 图的特征，之后 g 代表信息的交互，

$$\begin{cases} \mathbf{H}^{(n_i+1)} = f_3(\mathcal{G}(\mathbf{P}^{(n_i+1)}, f_3(\mathbf{H}^{(n_i)}))) \\ \mathbf{P}^{(n_i+1)} = f_3(\mathbf{P}^{(n_i)}) \end{cases}$$

f3 为 3*3 卷积，之后的 DDEM 包含注意力机制，SAB 相互并联行进，目的提取关键光谱空间信息，

$$\begin{cases} \mathbf{X}_1 = (\mathrm{S})\mathrm{W\text{-}MSA}(|\mathbf{X}_0|) + \mathrm{SAB}(|\mathbf{X}_0|) + \mathbf{X}_0 \\ \mathbf{X}_2 = f_{\mathrm{MLP}}(|\mathbf{X}_1|) + \mathbf{X}_1 \end{cases} \tag{9}$$

可由式 9 表示，之后进入到 ASS-Conv，生成双域动态核，分为两个路线，暂且称为上路线和下路线，上路线中，单卷积核产生一个 h*w*kˆ2 的空间核，下路线通过空间平均池化，线性层，激活函数生成光谱核 1*1*kˆ2*c，

$$\mathbf{X}_1(i,j) = \mathbf{X}_0(\Omega_k(i), j) \otimes |\mathbf{W}_{\mathrm{spa}}(i)| \otimes |\mathbf{W}_{\mathrm{spe}}(j)| \tag{11}$$

其中，Ωk (i)表示第 i 个像素周围的第 k ×k 个窗口(即第 i 个窗口)，Wspa(i)∈Rˆ（1×1×kˆ2）表示 Wspa 中第 i 个像素处的权值。Wspe(j)∈Rˆ(1×1×kˆ2)代表每组 Wspe 中所有第 j 个特征组成的权重，平均分为 k2 组。

由上述操作可以得到一组空间光谱潜在编码。（发现 kˆ2 在文中没有解释，根据笔者推断

k^2=c）

Spectral－Spatial Query Mapping

空间查询：采用正弦位置编码，后通过 MLP 得到初始的查询结果，最终空间查询结果为

$$\mathbf{H}_{\text{dec}}(x,y)=\sum_{\omega\in\Omega'_2(x,y)}\frac{A_\omega}{A}f_{\text{MLP}}(z^*_\omega,\text{Cat}(s',\boldsymbol{v}_L(x,y))) \quad (12)$$

类似于以往隐式表示。

光谱查询：

$$\mathbf{H}_{\text{hyb}}(b)=\text{Cat}(\boldsymbol{\Upsilon}_L(b),\mathbf{H}^{\uparrow\uparrow}(b),\mathbf{H}_{\text{dec}}). \quad (13)$$

即光谱的特征图像为空间正弦编码，b 波段的上采样图像，和空间查询的最终结果的连接，最终经由 MLP 还原。

DPINR

由于 PAN 和 HS 数据的特征往往具有空间的变异性，故在之前提出的 NaivePINR 的基础上，我们开发了 DynamicPINR 来处理空间变化。在 DynamicPINR 中，从两个方面考虑了空间的动态性:一是动态频谱-空间编码,产生动态潜码;二是动态查询映射,动态建立坐标-数据连接。

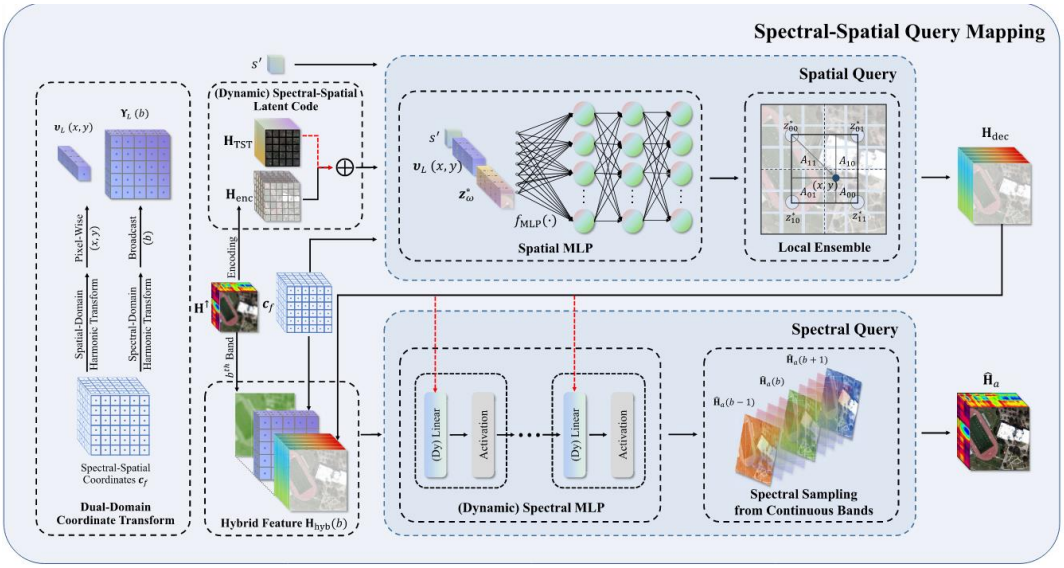
Dynamic Spectral－Spatial Encoding

Henc 为上述潜在编码， Φ 为获取全局信息的网络

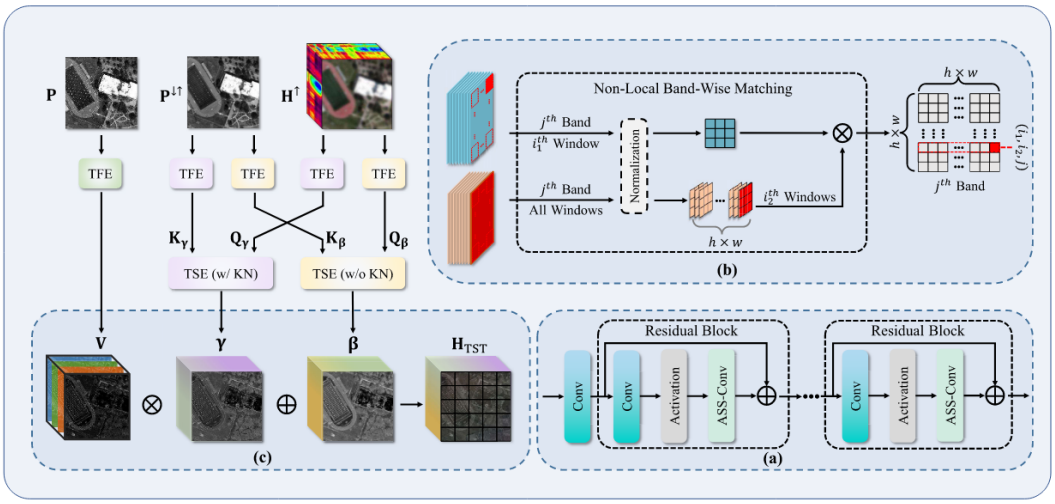
$$\mathbf{H}_{\text{dy}}=\mathbf{H}_{\text{enc}}+\varphi_{d1}(\boldsymbol{\theta}_{d1}) \quad (14)$$

Φ 为 TST 模块

具体动态网络如下所示



对于 TST 模块具体解释如下



首先 TFE 包含单个卷积层和残差网络，其次 TSE 建立了长范围 HS 和 PAN 之间的信息联系，为注意力的 KQ 机制，之后步骤详见文章，这样最终拿到动态的图像编码。

Dynamic Spectral－Spatial Query Mapping
动态查询映射加入了数据的自适应查询，

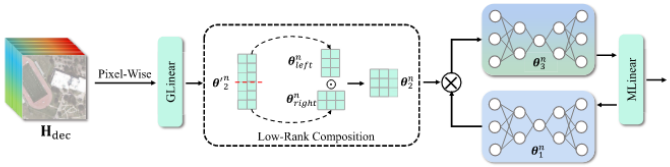


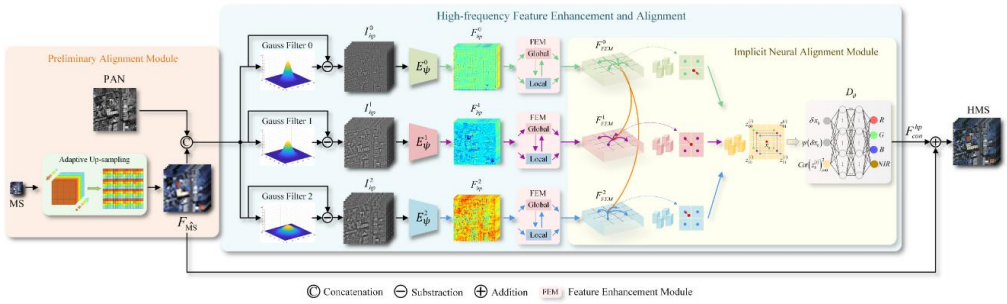
Fig. 6. Block diagram of a DyLinear layer in the spectral MLP.

每个 DyLinear 层由一个生成线性层（GL）和一个调制线性层（ML）组成，首先 GL 将 Hdec 生成一个临时参数，参数具有低秩特征，最终 b 波段的重建图像可由下式表示

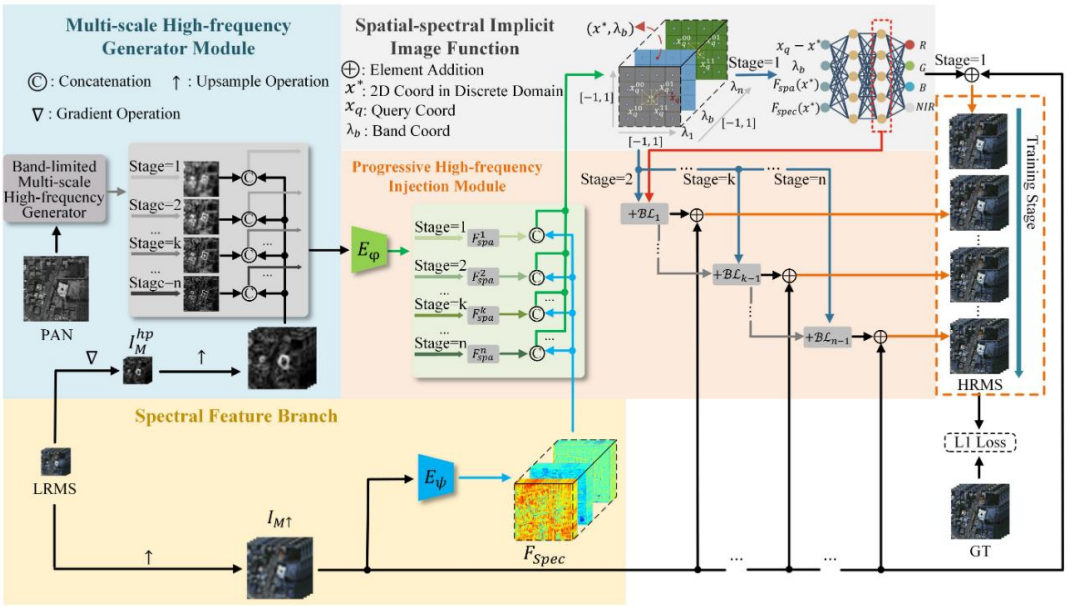
$$\widehat{\mathbf{H}}_a(b) = f^N_{\text{ml-}\delta}(\ldots(f^1_{\text{ml-}\delta}(\mathbf{H}_{\text{hyb}}(b);\boldsymbol{\theta}^1_3))\ldots;\boldsymbol{\theta}^N_3) \qquad (18)$$

总结：本片文章在阅读的困难性上个人认为超过前两篇，因为文章中出现大量符号代表特征图块，这让我在理解前后矩阵维度相符时造成了困难，但本篇文章深度地给出了输入图像的一步步编码的过程，并结合 QKV 注意力机制使得编码解码存在全局的关联，加深了对于 INR 的基本理解。

为什么要举以上三篇 INR 文章的例子，是因为个人理解，以上三篇对于模型的复杂度是逐层加深的，由（1）通过 MSI 直接训练网络参数，将图像复原为 HIS，再到后面（2）为实现图像超分在空间编码有了近一步操作，再到（3）加入不同特征图像之间的交互信息，使用全局的信息帮助还原图像，INR 的使用愈加灵活且广泛，当然由于个人理解能力有限，有些有关计算方面，以及具体代码实现还未能理解全面，因而会有错误和疏漏，恳请大家多多包涵。下面我不再详细分解具体的模型，只取一部分认为重要的点进行总结。



INR：为将 PAN 图的高频细节更好地注入，由于前面不同尺度的高斯模糊核导致在细节注入时有局部全局信息注入的差别，因此在空间对齐方面有着不小的困难，而 INR 在对齐方面有着优势，故选择。



这里便是一个比较常规的 INR

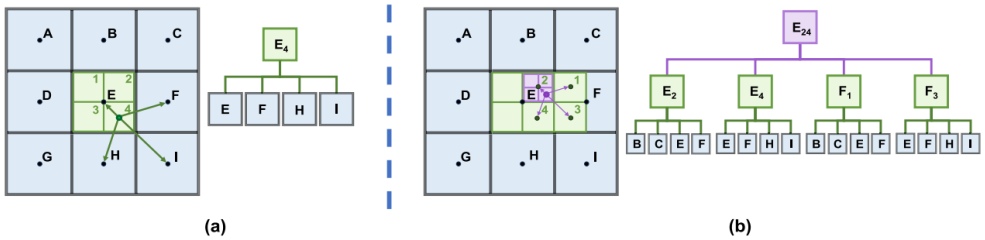


Fig. 1. (a) GINS can be neoterically considered as a quadtree structure. Taking the upsampling ratio ×2 times as an example, we divide the low-resolution image into nine pixels labeled as *A*–*I*. Then, based on the pixels, we assign clockwise numbers from 1 to 4 to the subpixels. Using the nearest-neighbor sampling method in GINS, it is evident that the subpixel E_4 is generated from the low-resolution pixels *E*, *F*, *H*, and *I*. (b) QIS is presented under upsampling ratio ×4 times, where we performed an upward growth operation on the quadtree and updated the new root node E_{24} to store the high-resolution subsubpixels.

这里是将预测像素值近一步扩大影响范围，图 b 所示，E24 这一点像素值将由更多节点的像素值所决定，是对上述提到像素值又周围四点决定的想法的近一步的优化。

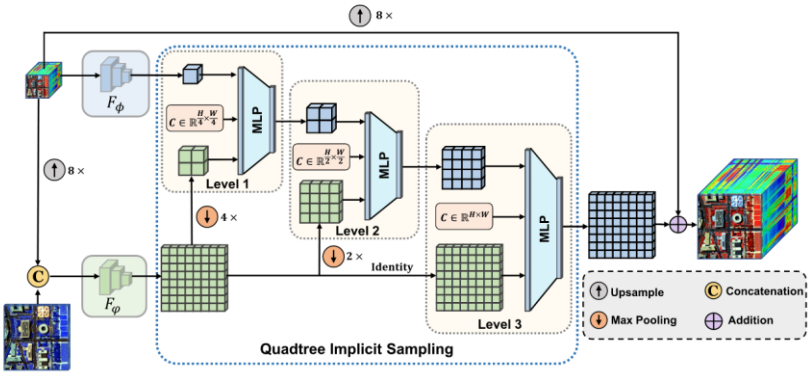


Fig. 3. For clarity, we establish the process of our generator at the upsampling ratio of ×8 and take HR-MSI and LR-HSI as inputs. First, we perform interpolation on the LR-HSI to achieve the desired size. Then, we concatenate the upsampled LR-HSI with the HR-MSI. Next, we use networks F_φ and F_ϕ to learn the latent codes of the concatenated LR-HSI and HR-MSI, respectively. Finally, we employ the proposed QIS method to fuse the two codes together and add them with upsampled LR-HSI to generate the final output.

这是具体结构

4. 对于 INR 应用的看法

   INR 可以有效地逼近复杂函数，可以生成连续光滑的输出，特别在图像超分以及 HSI,MSI 任务中表现出色，然而，其训练复杂性、计算成本高、稳定性问题和解释性差等缺陷仍需克服。下面要多加关注图像 encoding decoding 的过程，如何更好地关注到图像高频细节的注入，以及除 MLP 之外的其他形式。