理解 Stable Diffusion UNet 网络

2024-5-26

在前面的学习中，我们把 SD UNet 网络当成黑盒，不太影响对图片生成大致原理的理解，但在继续学 SD 的过程中，发现 ControlNet、T2I-Adapter、IPAdapter 等这些技术，都是在原 SD 网络模型上以各种方式对网络做修改叠加，要理解这些技术，还是得先了解下 SD UNet 网络结构的一些细节，不然看得很费劲。

SD 模型构成

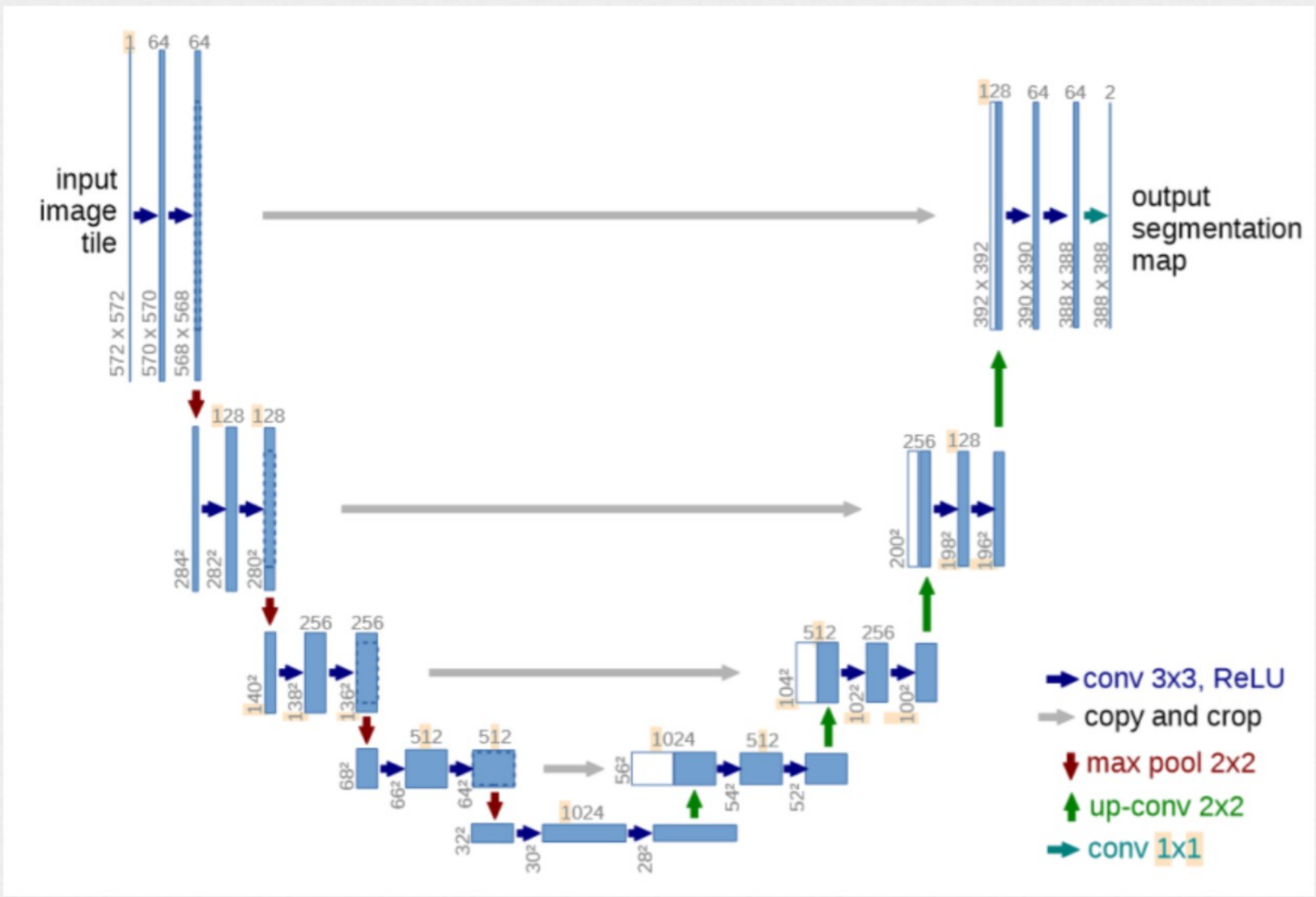
从之前的学习我们知道，Stable Diffusion 模型里包含了三个组件：CLIP、VAE、UNet，这三个组件的参数和大小分布(来源)：

组件	参数个数	文件大小	占比
CLIP	123,060,480	492 MB	12%
VAE	83,653,863	335 MB	8%
UNet	859,520,964	3.44 GB	80%
Total	1,066,235,307	4.27 GB	100%

整个生图的核心网络就是 UNet。UNet 最初是用于医学图像分割任务而提出来的，但它的特性展现了在图像其他领域的应用潜力，后续经过扩散模型的改进，很好应用在了图像生成上，所以 Stable Diffusion 的 UNet 实际上在原 UNet 网络架构上做了一些改造。

基础 UNet 网络

我们先来看看原 UNet 网络架构：



1. 左边输入图片，经过整个网络处理，右边输出同尺寸图片。（原 UNet 网络用于医学图像识别分割，所以图上右边标的输出是一张同尺寸分割图。SD 这里的输出是降噪图）

2. 左边下采样（也可以称为编码器），右边上采样（也可以称为解码器），一张图片经过一层层下采样计算，尺寸逐渐减小（图中的网络是减小到32×32），再经过右边层层上采样，恢复到原尺寸。那这里下采样和上采样的作用是什么?
3. **下采样**，是使用某种计算方式让更小的数据表示整张图片，这更小的数据代表了对这张图片高纬度的描述，而不是像素级细致的描述。
    1. 越小的数据对图片的表示和描述越宏观，有利于捕捉图片的语义特征。
    2. 例如一张猫在屋子前玩耍的地图，原图能看清所有细节，但因为细节太多，模型想要知道图里有猫和屋子，得把每个像素组合运算才行，但下采样到最小，最宏观的猫和屋子就容易识别。
4. **上采样**，是让图片的宏观小尺寸表示恢复成原图片尺寸。
    1. 比如对于图片分割（把图片上的物体分割出来），我们在下采样后的小数据量的高维表示里识别了图片的主体、边缘，最后还是要转回在原尺寸图片上表示，不然识别了也没用。
    2. 那不断下采样过程中肯定把图片细节都丢失了，再上采样，怎么可能还原图片细节？那就要说到跳跃连接（skip connection）了。
5. **跳跃连接**，也就是并不是顺着网络的方向连接，而是跳过原网络方向，跳着连接传输信息。说得有点拗口，看图很容易理解，就是图上中间的几条灰色箭头。
    1. 原网络连接方向是图片输入→下采样各节点→上采样各节点→输出图片这个链路，就是图中U字型的路径。
    2. 在这个路径之外，左边的下采样的每一层，都额外连接到右边上采样对应的层上面，将两个网络进行拼接。
    3. 上采样每一层，都在拼接了左边下采样对应层的数据后，再一起作为下一层上采样的输入。
    4. 为什么这样做，很容易理解，左边的每一层网络都保留了图片不同程度的细节，右边的每一层因为是上采样过来的，只有宏观信息，没有图片细节，那把左边图片细节信息拼接过去，右边这个网络宏观特征和微观细节都具备了，每一层都有不同程度的对图片的宏观语义理解和微观细节，就能做各种事情了，包括图片分割、语义生成图片。

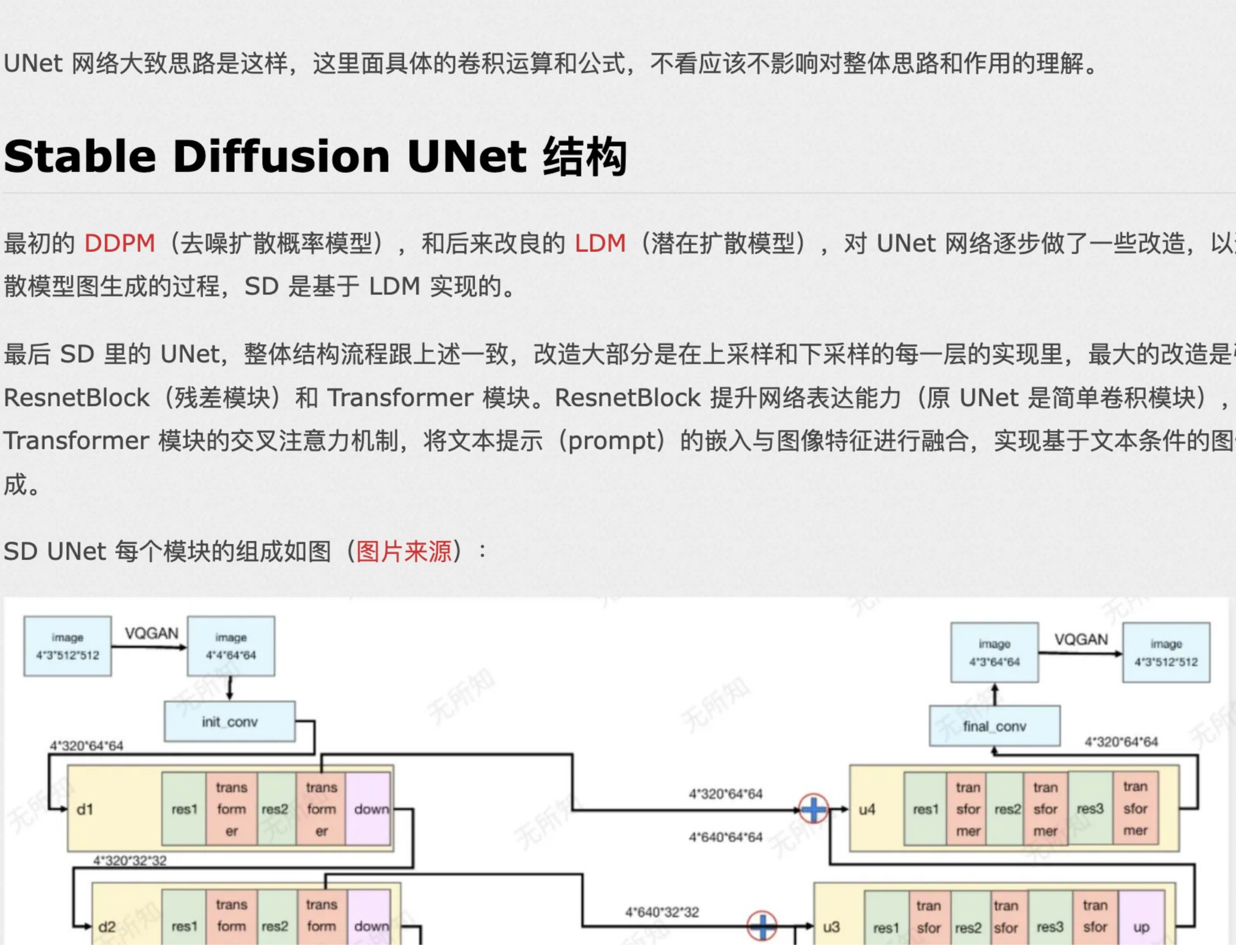
UNet 网络大致思路是这样，这里面具体的卷积运算和公式，不看应该不影响对整体思路和作用的理解。

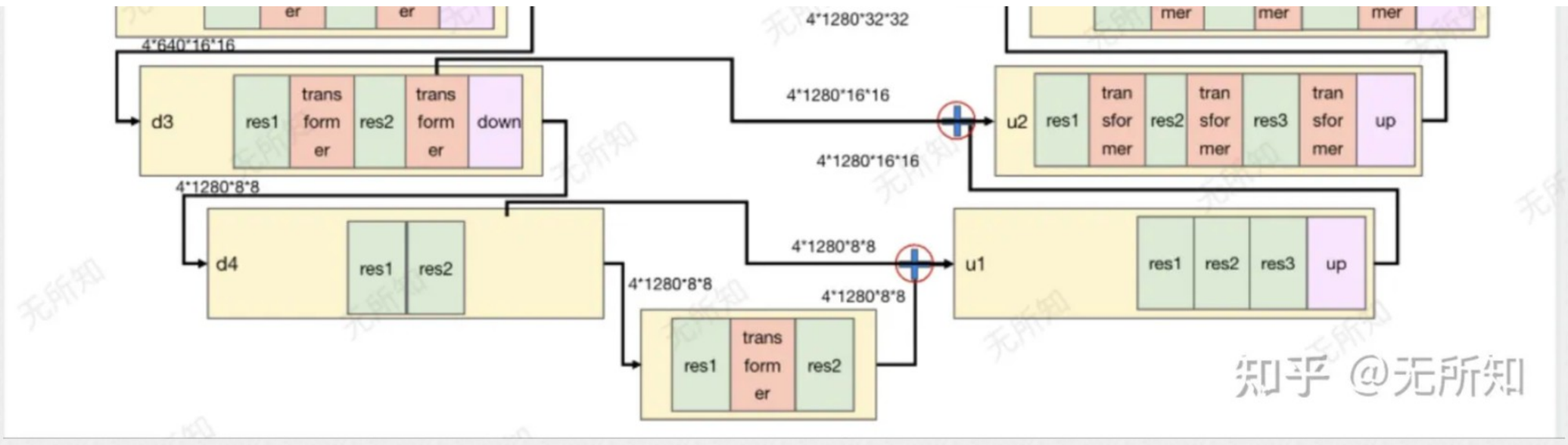
Stable Diffusion UNet 结构

最初的 DDPM（去噪扩散概率模型），和后来改良的 LDM（潜在扩散模型），对 UNet 网络逐步做了一些改造，以适合扩散模型图生成的过程，SD 是基于 LDM 实现的。

最后 SD 里的 UNet，整体结构流程跟上述一致，改造大部分是在上采样和下采样的每一层的实现里，最大的改造是引入了 ResnetBlock（残差模块）和 Transformer 模块。ResnetBlock 提升网络表达能力（原 UNet 是简单卷积模块），而 Transformer 模块的交叉注意力机制，将文本提示（prompt）的嵌入与图像特征进行融合，实现基于文本条件的图像生成。

SD UNet 每个模块的组成如图（图片来源）：

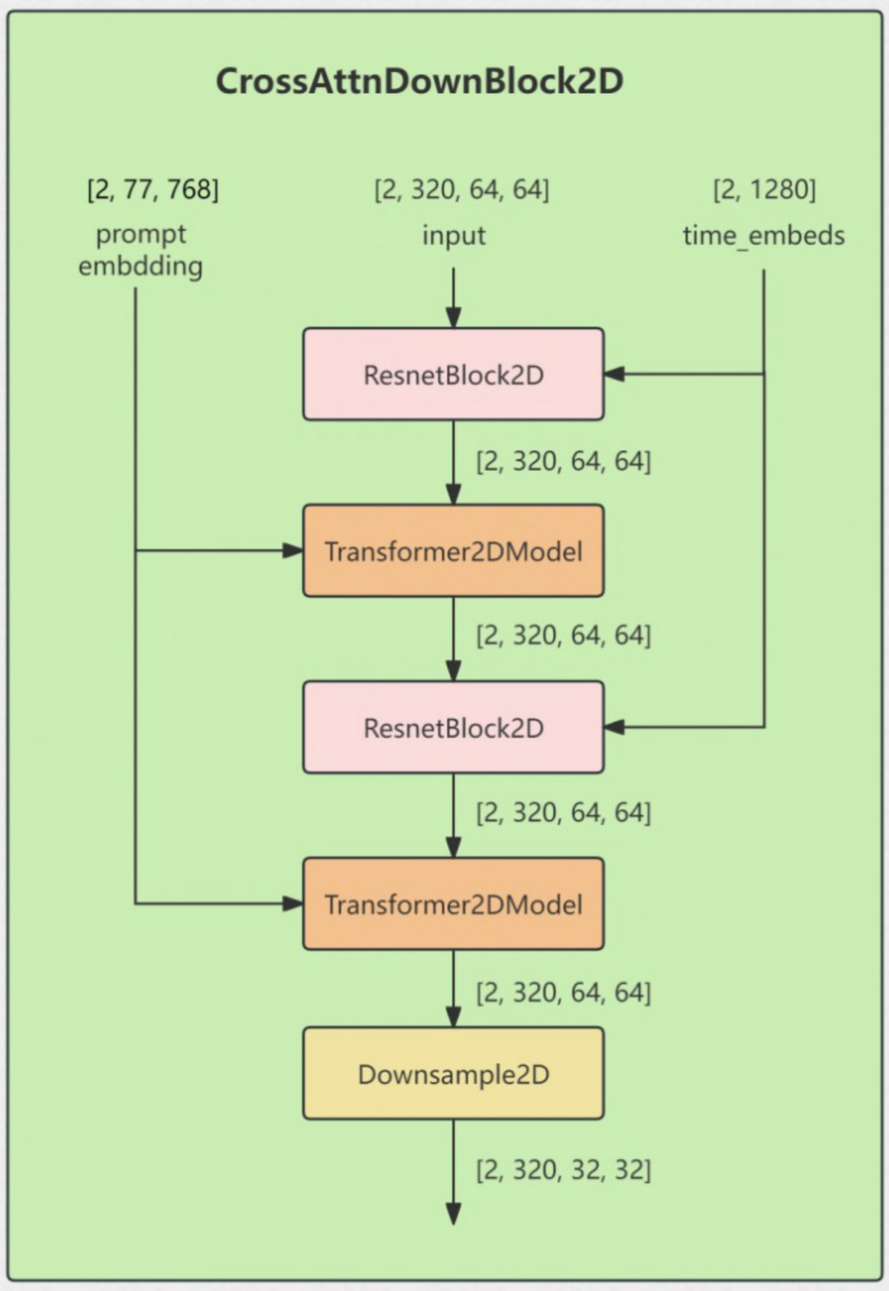




左边下采样每层由2个残差模块和2个Transformer模块连接组成，右边上采样是各3个，中间层是2个残差模块和1个Transformer模块。（高维的d4和u1没有接入Transformer模块，原因不明，可能是试过加入后效果不佳，在高维这里加入 Prompt 交叉注意机制，文字权重太大？）

细分模块结构

里面每一块具体的结构这篇文章画得很详细，摘录学习一下。我们拿其中一个下采样模块看看：

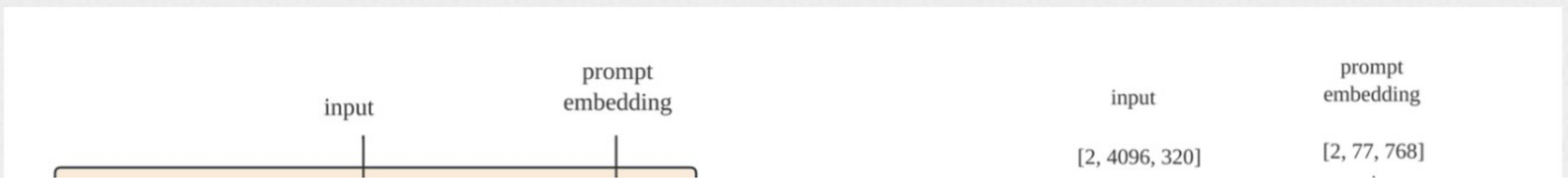


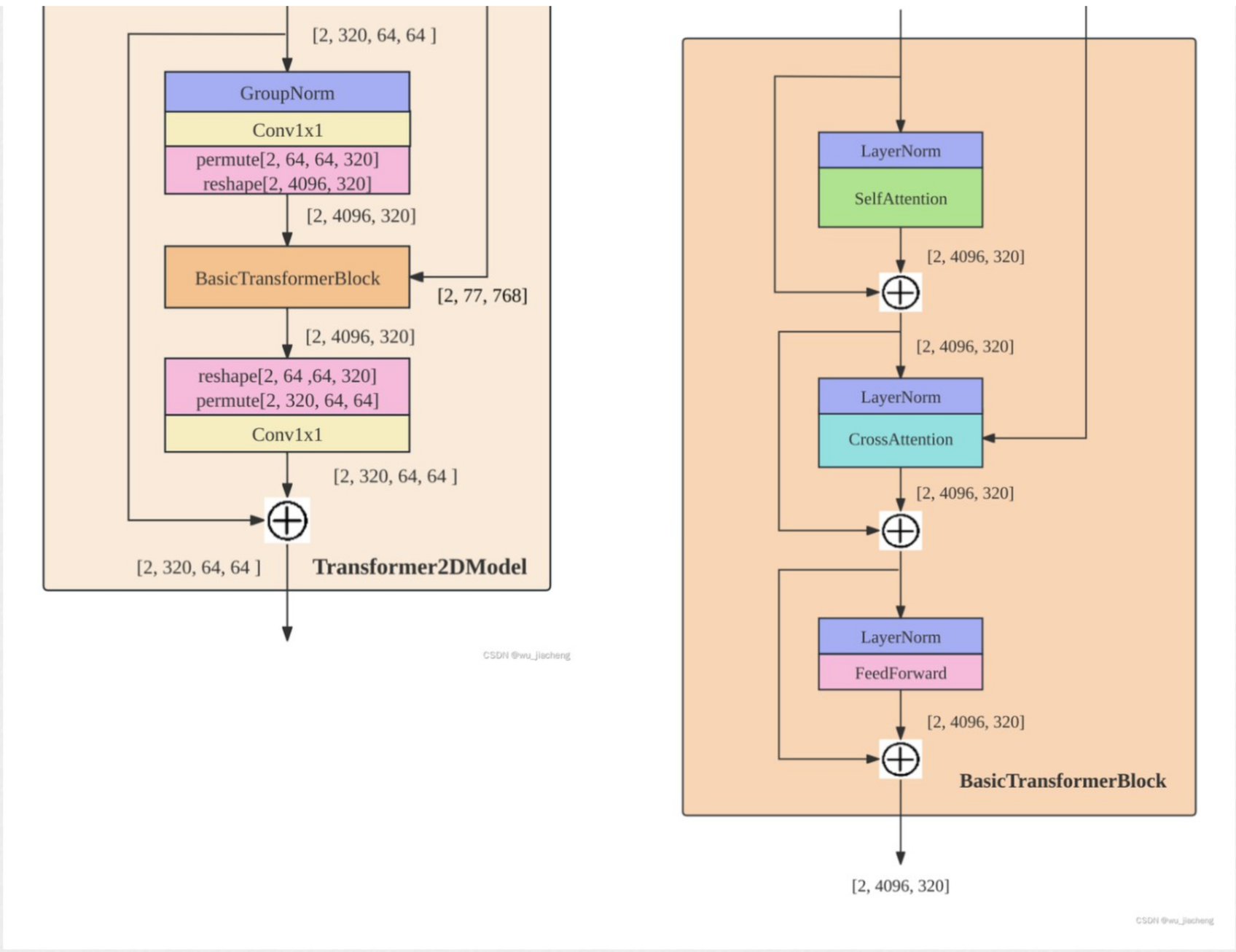
两个残差模块，两个Transformer模块。这图表示了 SD 生图的三个输入：input（噪声图）、prompt_embdding（文字 Prompt）、time_embdding（步数）在这几个模块的流转和处理。这里每一个小模块处理完后，输出的可以近似认为都是一个预测的噪声图的数据表示。

残差模块的输入输出 噪声图+步数 → 预测噪声图，Transformer 模块的输入输出是 噪声图+ Prompt → 预测噪声图。

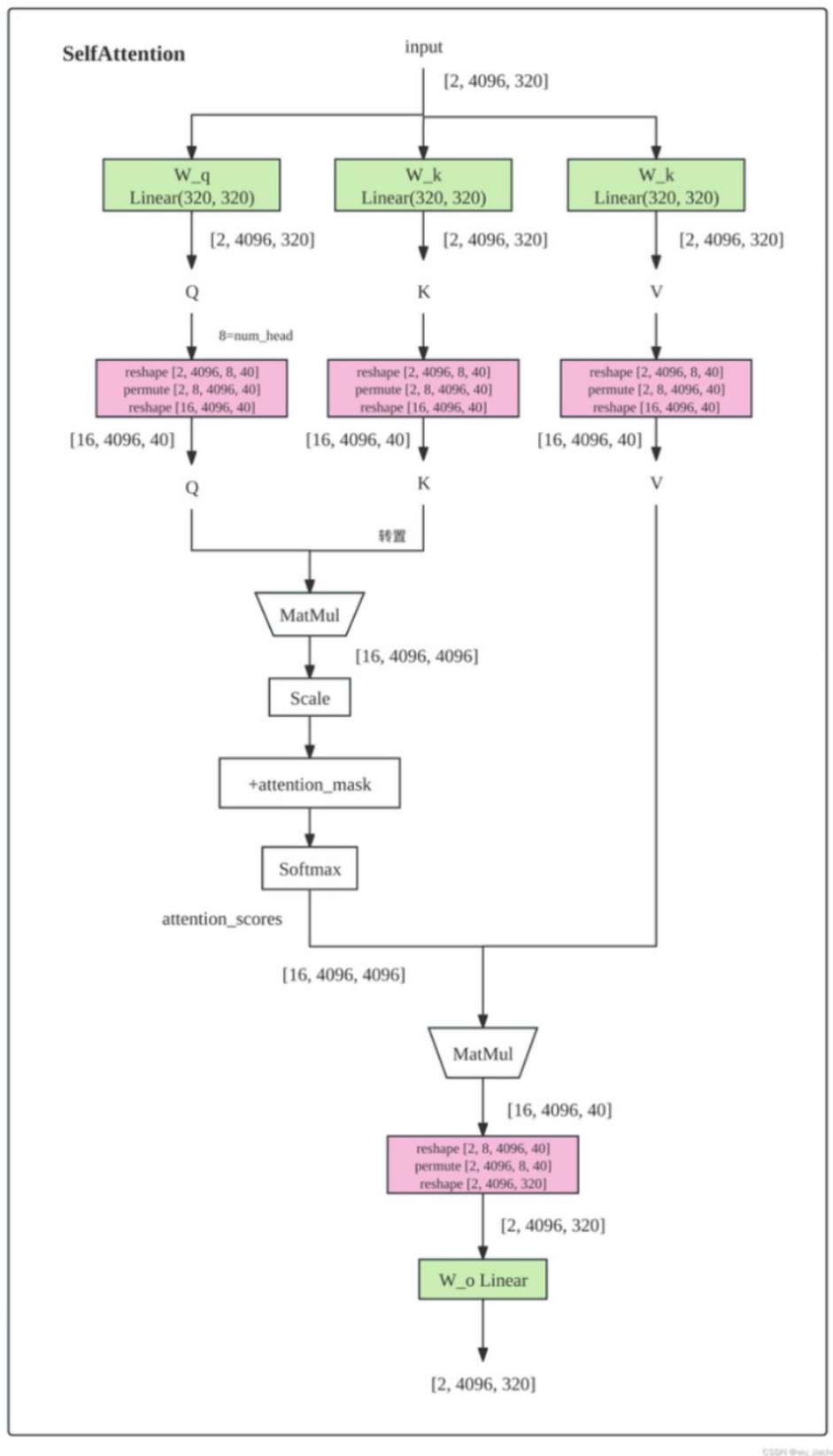
Transformer 模块

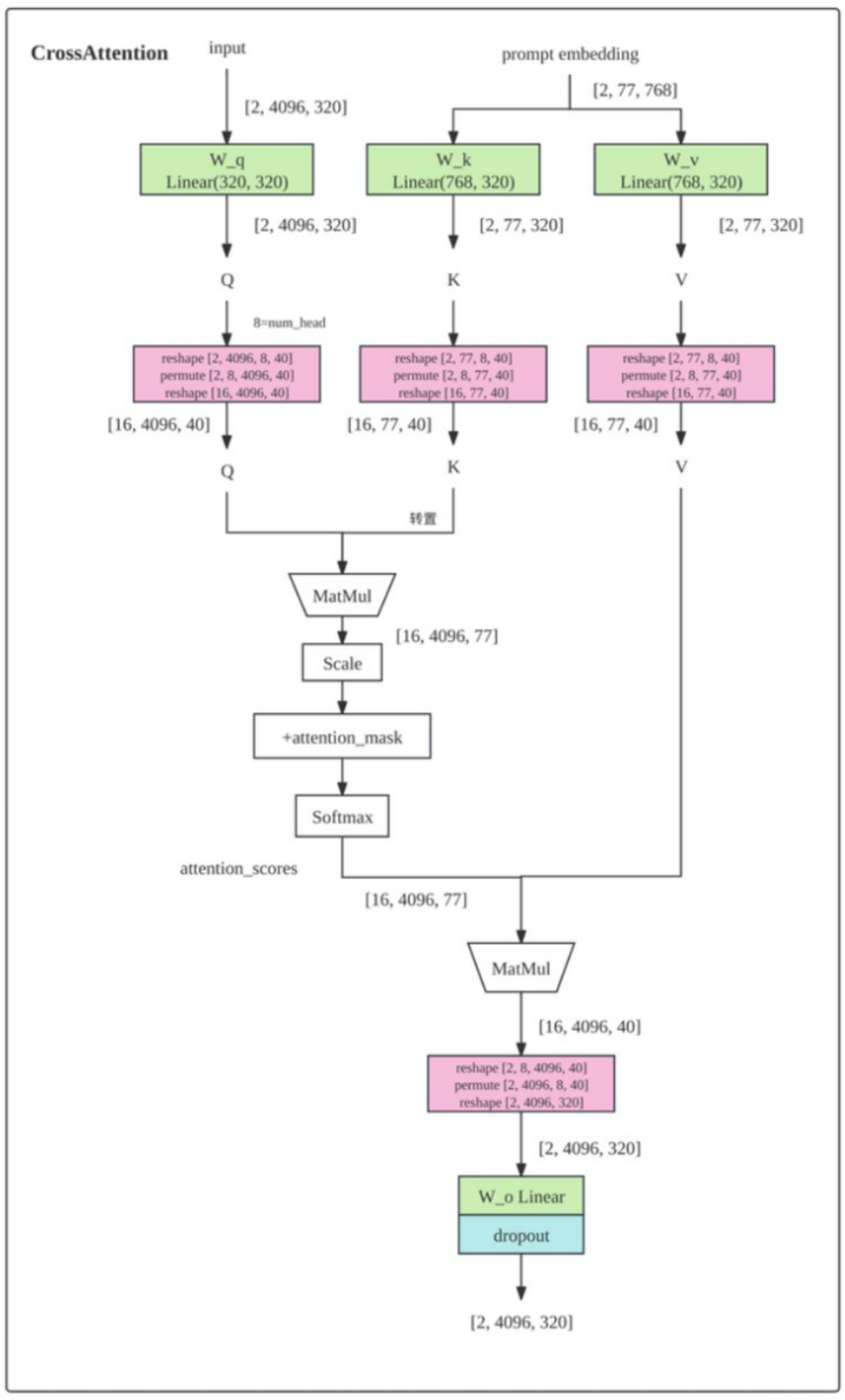
再细看一下 Transformer 模块，Transformer 模块由下图所示好几个部分组成，最主要的是 自注意力模块（SelfAttention）和交叉注意力模块（CrossAttention）：





展开看看这两个模块：





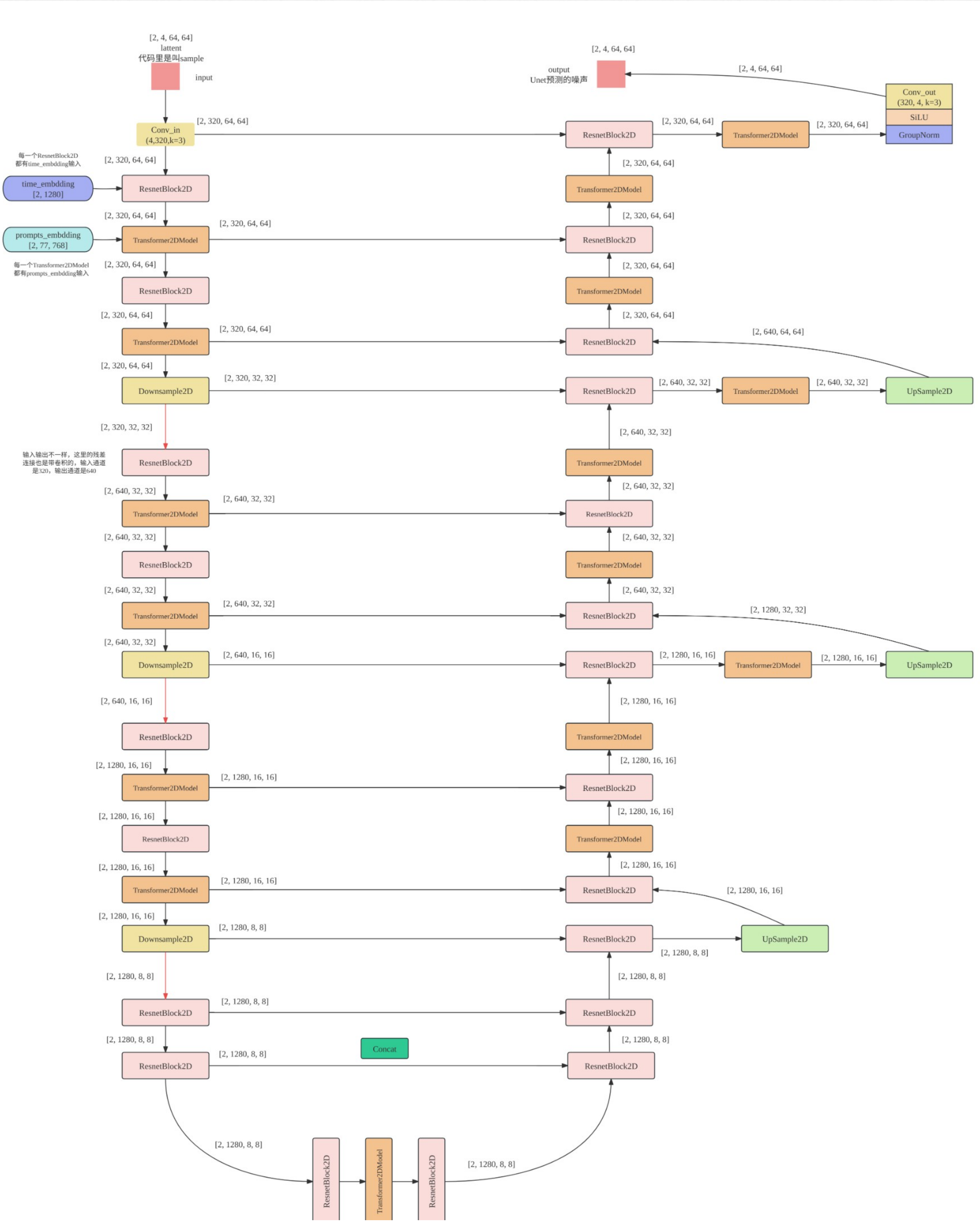
自注意力模块，Transformer 结构里的 QKV 输入都是图片特征（上一层的处理结果，就是降噪图的特征），这样做可以让模型获得包含整个输入图像的感受野，捕捉图片特征中不同位置之间的关系， 全局感受力是 Transformer 架构的特点。

交叉注意力模块，它的作用是融合不同模态的输入，在这里就是融合噪声图和文本特征，Q的输入是图片特征，KV的输入是文字 prompt_embedding，让图片特征可以关注到文字输入，根据注意力权重调整图片的生成方向。文字 prompt 在整个Transformer模块中只作用在交叉注意力这部分里。

Transformer 的机制原理、QKV的含义，是另一个比较大的话题，可以先看看网上其他相关讲解，比如这篇，后续再细拆深入。

回顾

关键几个模块的组成了解了，再回到整个UNet的构成：



现在通过这些结构图，可以大致看到 UNet 网络里的整体处理流程，以及关键模块的作用，经过这些模块的逐个叠加，组合成一个个采样模块，再组合成 UNet 网络架构，完成整个生图运算。

这里面还有很多需要深入学习的点，当前先了解到这个维度，已经可以帮助大致理解后续 ControlNet 等网络的机制原理。