我想建造一个模型，用来提取codebook token。首先初始化一个16X64的codebook或者叫特征向量集C1，然后对他采用自注意力机制重新分配得到C2,形状是16x128,C2经过batchsize次的重参数化采样，得到batchsize个C3，shape是（batchsize，16，128），然后经过一个合理的生成器G1,分16个区域生成128X128的图片F1，现在F1数据形状是（batchsize，128，128）.然后真实数据P1

,也是（batchsize，128，128）。P1与F1，两者经过相同的抽取器E1,E1的目的是将其先划分16个区域，然后每个区域用CNN将其还原为（batchsize，16，128）向量，F1还原回的是RF1,P1还原回的是RP1,经过一个判别器C1，进行对抗性训练，判别器是要把RF1的16个向量都判断为伪造，Rp1的16个向量都判断为真实。而之前的生成器其实就是要骗过C1，让他把RF1

的16个向量也判断为真实。然后，把RF1和RP1的16个向量，用最小L2距离，比照C2 16个向量进行替换，CP1替换的过程中计算一个替换损失L\_RP1,替换完之后的是RP2,然后对RP2经过生成器G1,生成的GRP2,形状是（batchsize，128，128），与对应的图片P1做重构损失，记作L\_R\_GRP2。RF1也经过替换生成RF2，然后相同的生成器G1,生成GRF2。GRF2,与GRP2共同经过相同的抽取器E1，E1的目的是将其先划分16个区域，然后每个区域用CNN将其还原为（batchsize，16，128）向量，GRF2还原回的是RGRF2,CRF2还原回的是RCRF2,经过一个判别器C1，进行对抗性训练，判别器是要把RGRF2的16个向量都判断为伪造，RGRP2的16个向量都判断为真实,同样，对抗方式是之前的生成器G1肯定是让 RGRF2被C1判断为真实.