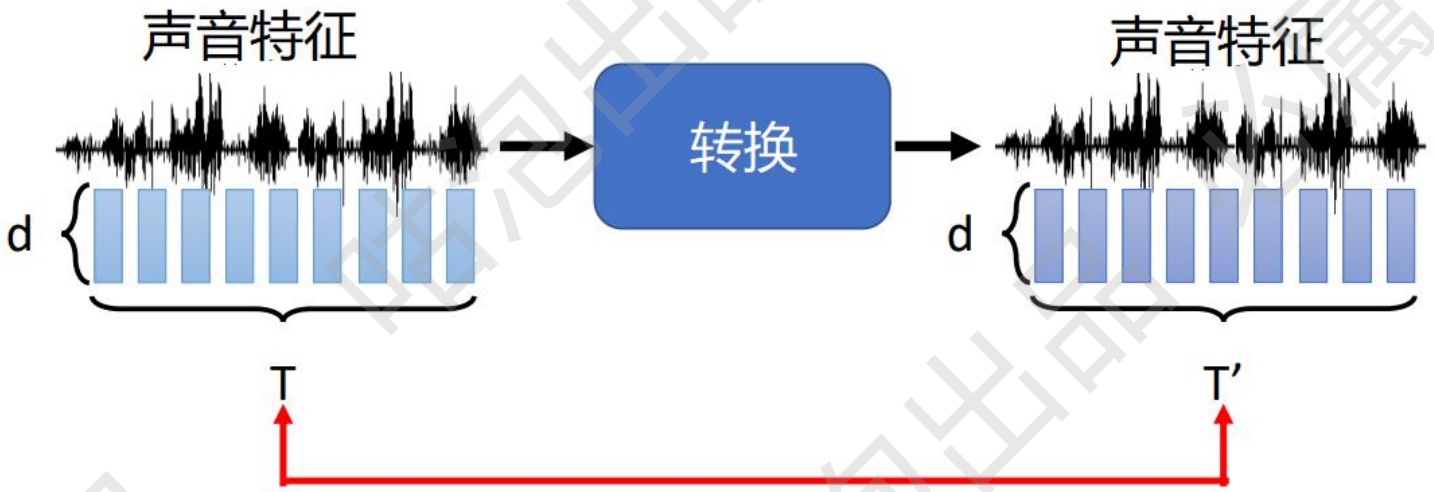
Stargan-vc2

- 变声器

  - 变声器的工作原理是什么呢?

  - 其实就是把语音特征进行转换，只不过内容不能变!



Stargan-vc2

- VC：Voice Conversion
  - 如何构建一个变声器呢？思想跟stargan差不多，细节完全不同
  - 需要输入什么？1.声音数据；2.标签编码；
  - 整体来说还是GAN模型，主要解决数据特征提取，网络模型定义
  - stargan-vc2是升级版，前身还有cyclegan-vc和stargan-vc

Stargan-vc2

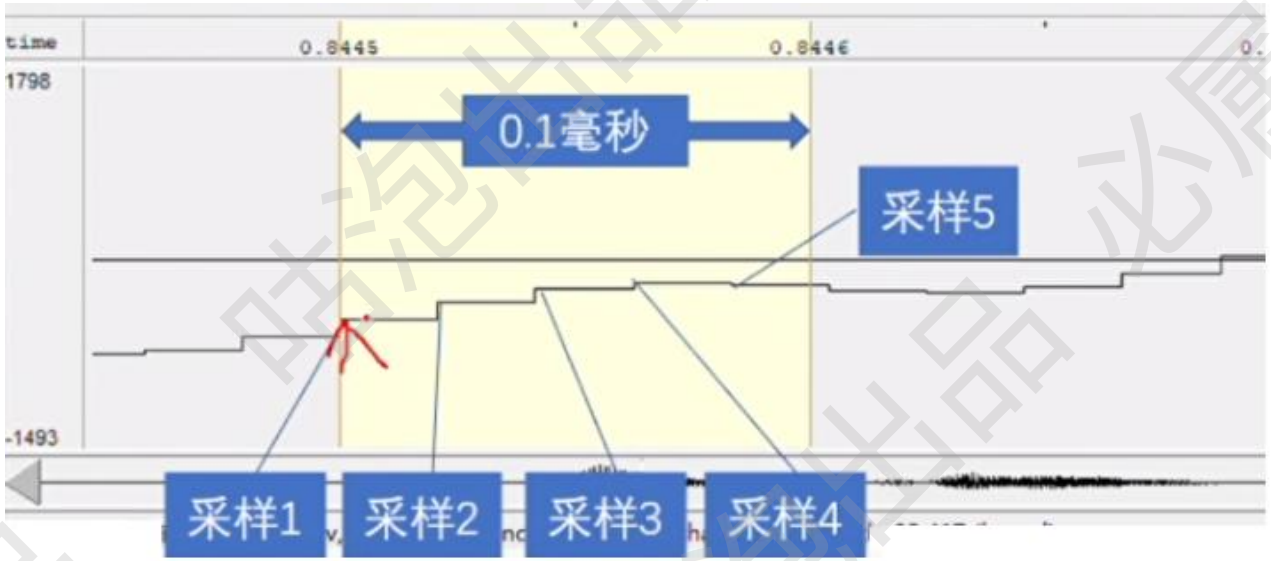
- 输入数据
  - VCC2016和VCC2018（这个数据相对较小），也可以用其他的
  - 4个人的声音数据，相当于4个domain，他们之间相互转换
  - 论文中选择的特征为：MCEPs；log F0；APs
  - 输入特征为：batchsize*1*35*128（35为特征个数，128为指定特征维度）

Stargan-vc2

- 输入数据

  - 频率：每秒钟波峰所发生的数目称之为信号的频率，用单位千赫兹(kHz)表示

  - 0.1毫秒完成4.8次采样，则1秒48000次采样，采样率48KHZ



Stargan-vc2

- 预处理
  - 16KHZ重采样（经验值，和论文一致）
  - 预加重：补偿高频信号，让高频信号权重更大一些，因为它信息多
  - 分帧：类似时间窗口，得到多个特征段
  - 论文中并没有详细介绍预处理内容，源码中按照通用套路来做的

Stargan-vc2

- 特征汇总

  - 基频特征（F0）：声音可以分解成不同频率的正弦波，其中最低的那个

  - 频谱包络：语音是一个时序信号，如采样频率为16kHz的音频文件（每秒包含16000个采样点）分帧后得到了多个子序列，然后对每个子序列进行傅里叶变换操作，就得到了频率-振幅图（也就是描述频率-振幅图变化趋势的）

  - Aperiodic参数：基于F0与频谱包络计算得到

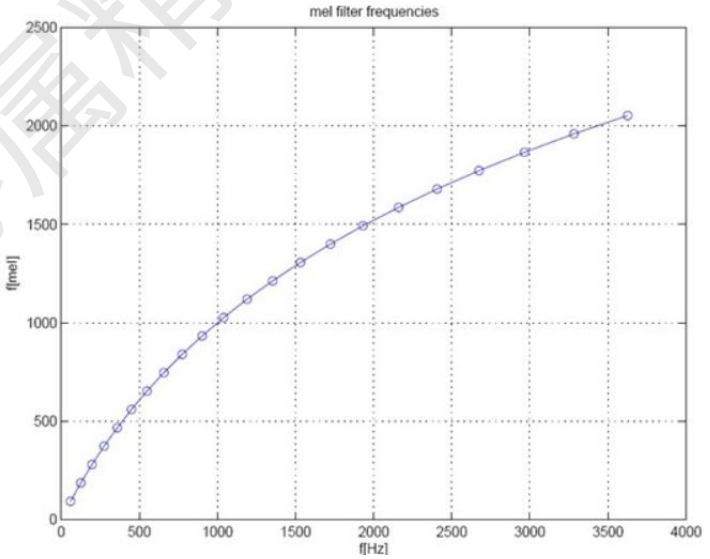
Stargan-vc2

- MFCC

  - 梅尔倒谱系数：
    流程：连续语音--预加重--加窗分帧--FFT--MEL滤波器组--对数运算--DCT

  - 通俗解释：FFT之后就把语音转换到频域
    MEL滤波器变换后相当于得到更符合人类听觉的效果：

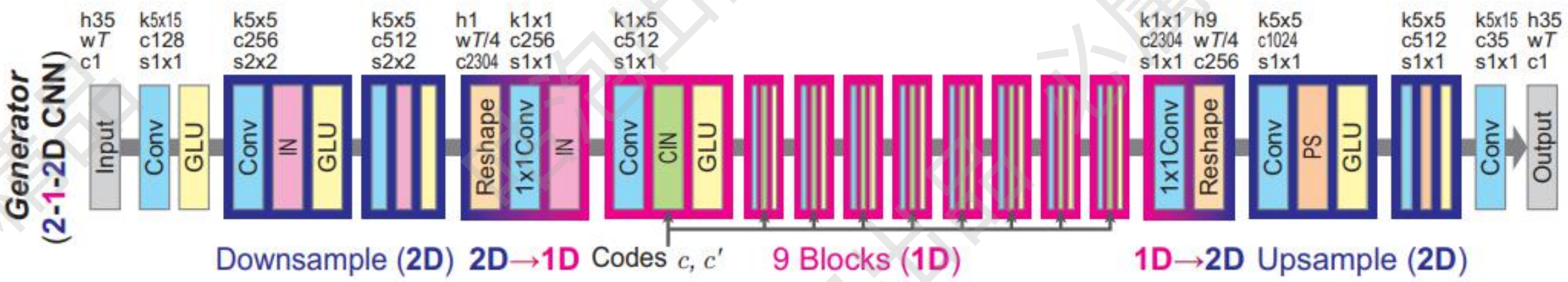
$$f_{mel}(f) = 2595 \cdot \log\left(1 + \frac{f}{700Hz}\right)$$



  - 最后DCT相当于提取每一帧的包络（这里面特征多）

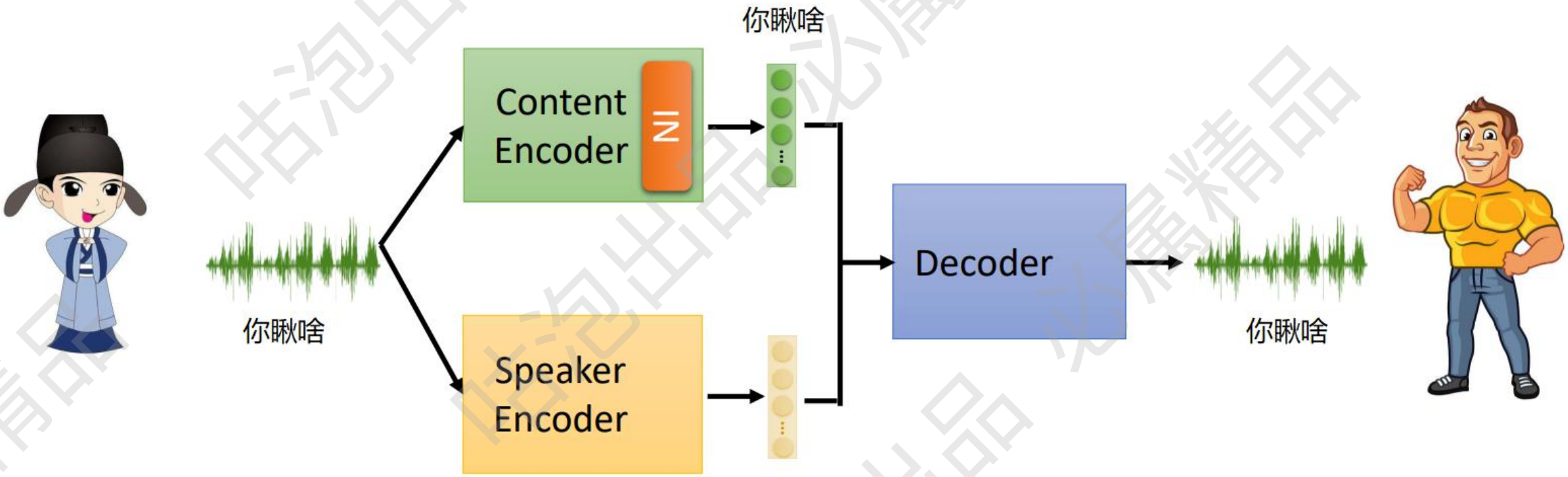
Stargan-vc2

- 网络架构
  - 生成器：输入就是提取好的特征，输出也就是特征
  - 感觉就是编码-解码的过程，其中引入了IN和GLU单元



Stargan-vc2

- 语音数据包含的成分

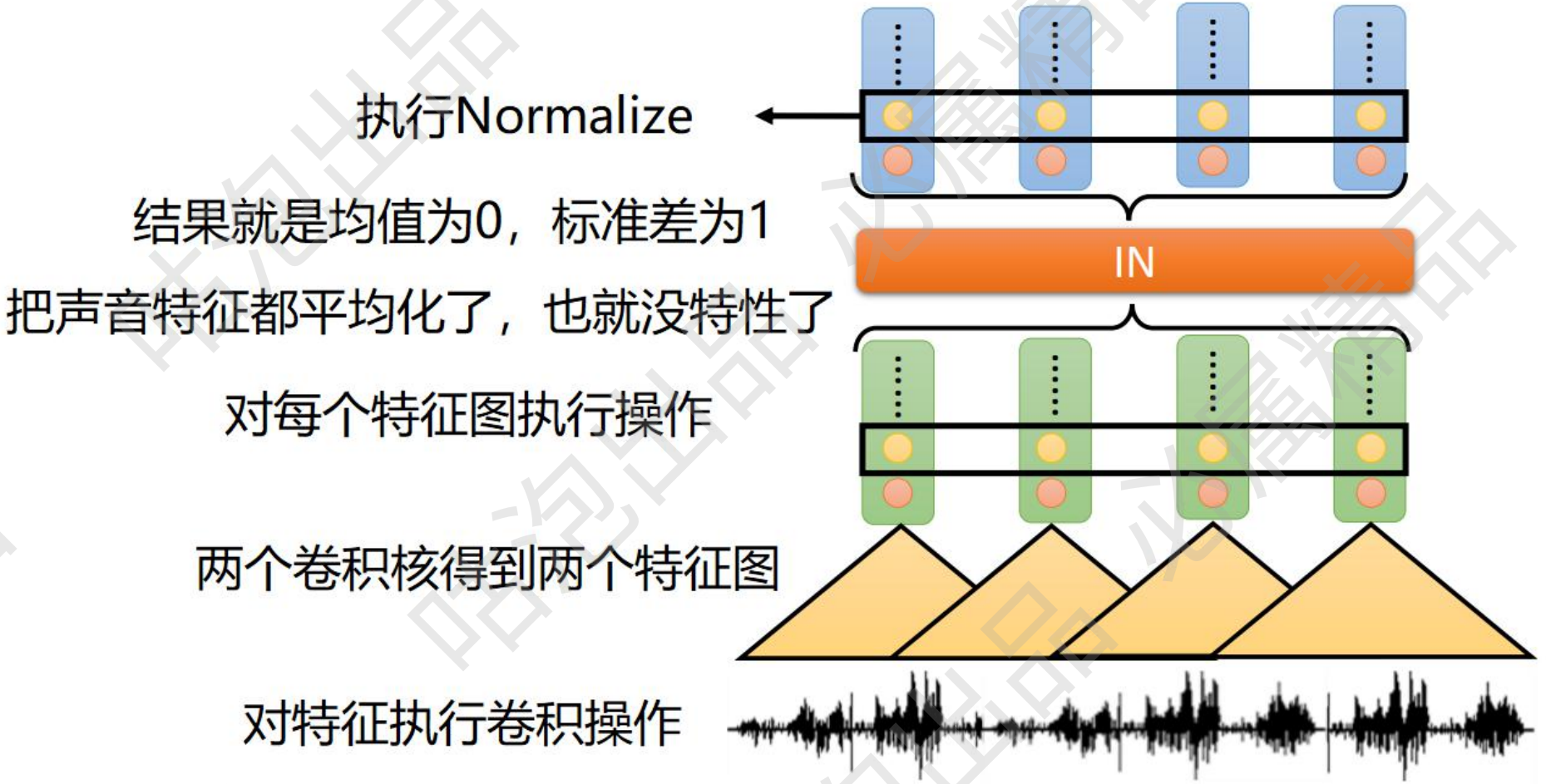


Stargan-vc2

- Instance Normalization
  - 变声器虽然把咱们动静给改了，但是内容没变吧！
  - 编码时如何保留住原始内容呢？这就得去掉声音中特性的部分
  - 解码时如何放大个性呢？还是需要再处理解码特征
  - Instance Normalization与Adaptive Instance Normalization

Stargan-vc2

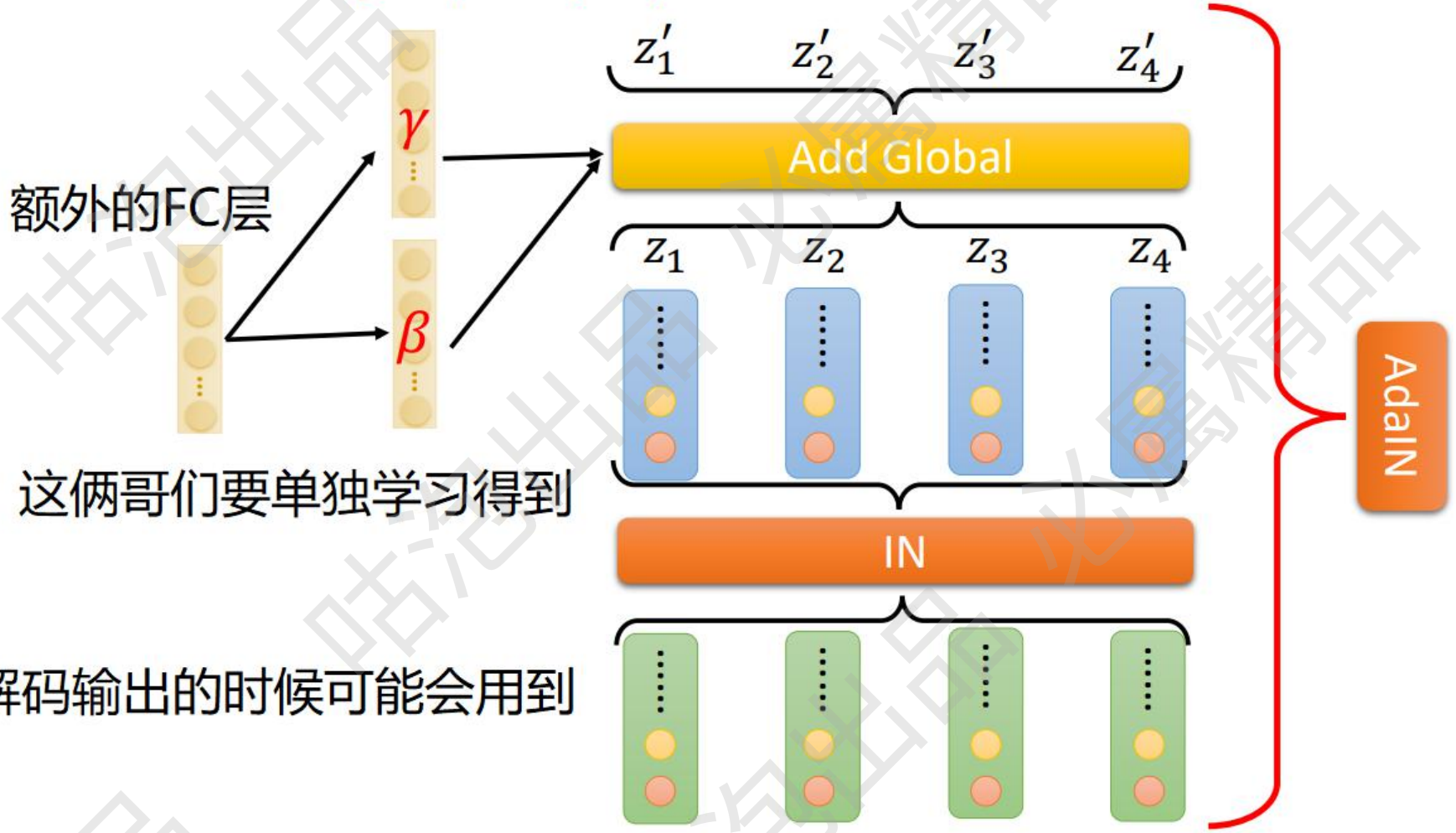
- Instance Normalization



Stargan-vc2

- AdaIn

$z_i' = \gamma \odot z_i + \beta$ 跟BN有些类似，做了补偿又

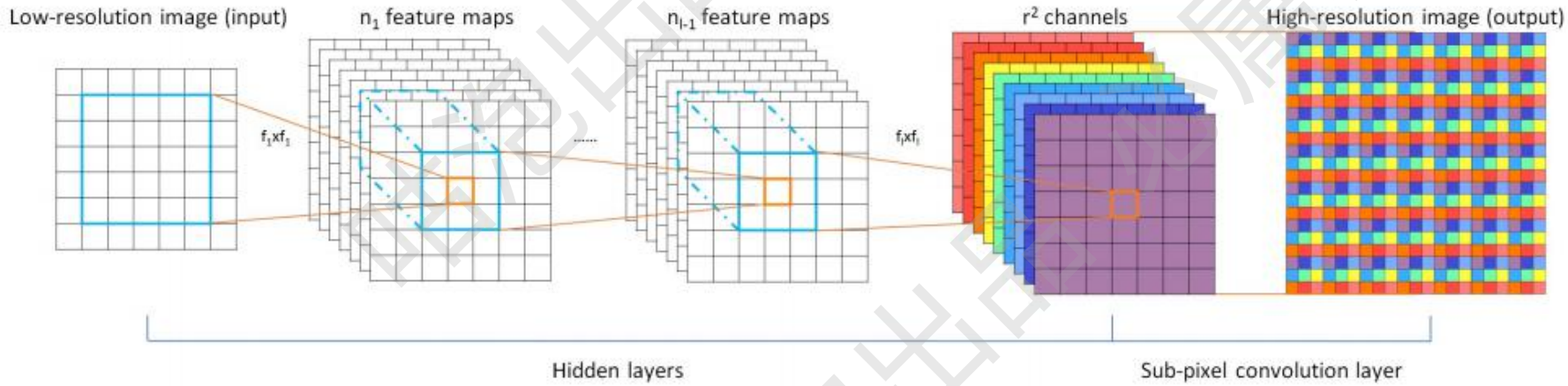


Stargan-vc2

- 小细节

  - 上采样与下采样：都是老路子，stride=2来下采样，反卷积来上采样

  - PixelShuffle（Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network）



Pixelshuffle会为 (*，r^2 x C, H, W) 的Tensor给reshape成 (*，C, rH, rW)

Stargan-vc2

- 判别器

  - GSP：global sum pooling：一个特征图压缩成一个点，batch*512*h*w压缩成batch*512

  - 标签通过embeding编码成512维特征(batch*512)，内积得到batch个判别结果