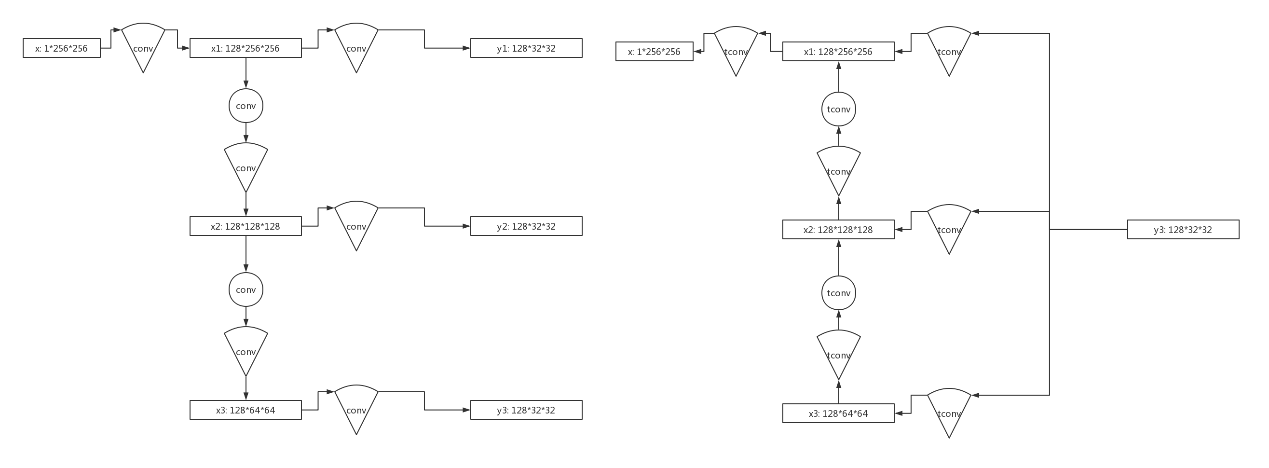
5.py结构

三角形conv表示的是降采样conv，圆形的则是没有降采样

编码器的三角形conv后会有一个gdn

解码器的三角形conv前会有一个gdn

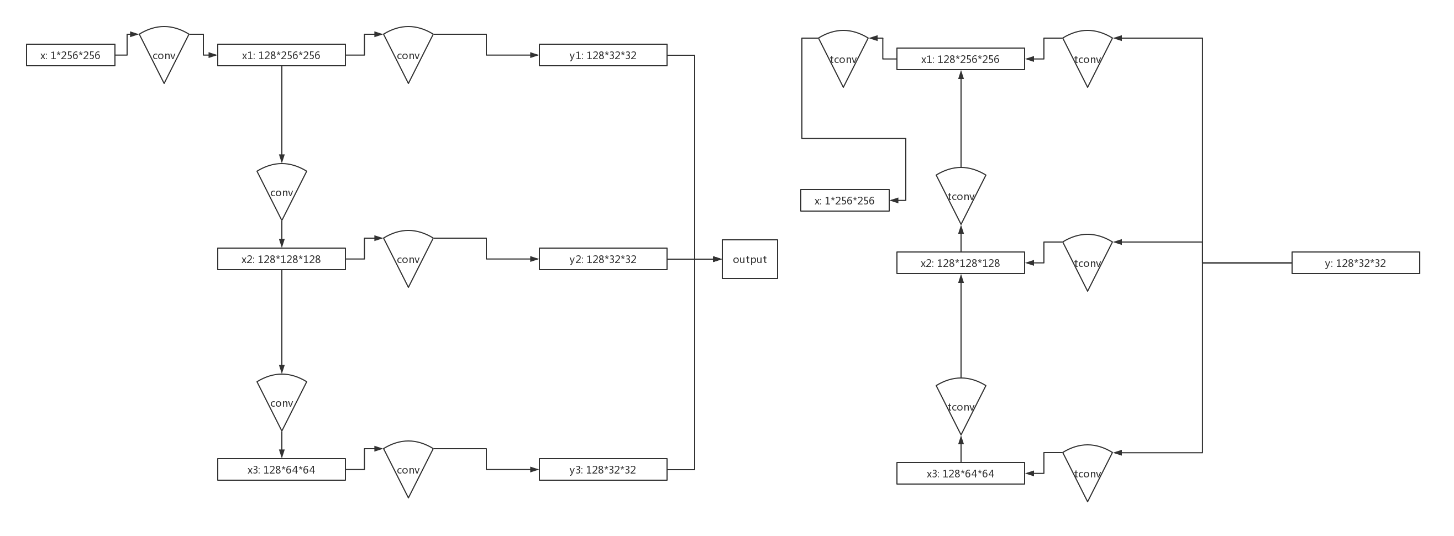


5\_1.py的改进为，增加了圆形conv层数量，但效果反而不如5.py

5\_2.py将竖直方向的加号改为减号

x1A = x1  
x1 = F.leaky\_relu(self.conv1\_0(x1))  
x1 = F.leaky\_relu(self.conv1\_1(x1))  
x1 = F.leaky\_relu(self.conv1\_2(x1))  
x1 = F.leaky\_relu(self.conv1\_3(x1))  
x1 = x1 - x1A

5\_3.py在5.py基础上直接去掉了圆形conv层



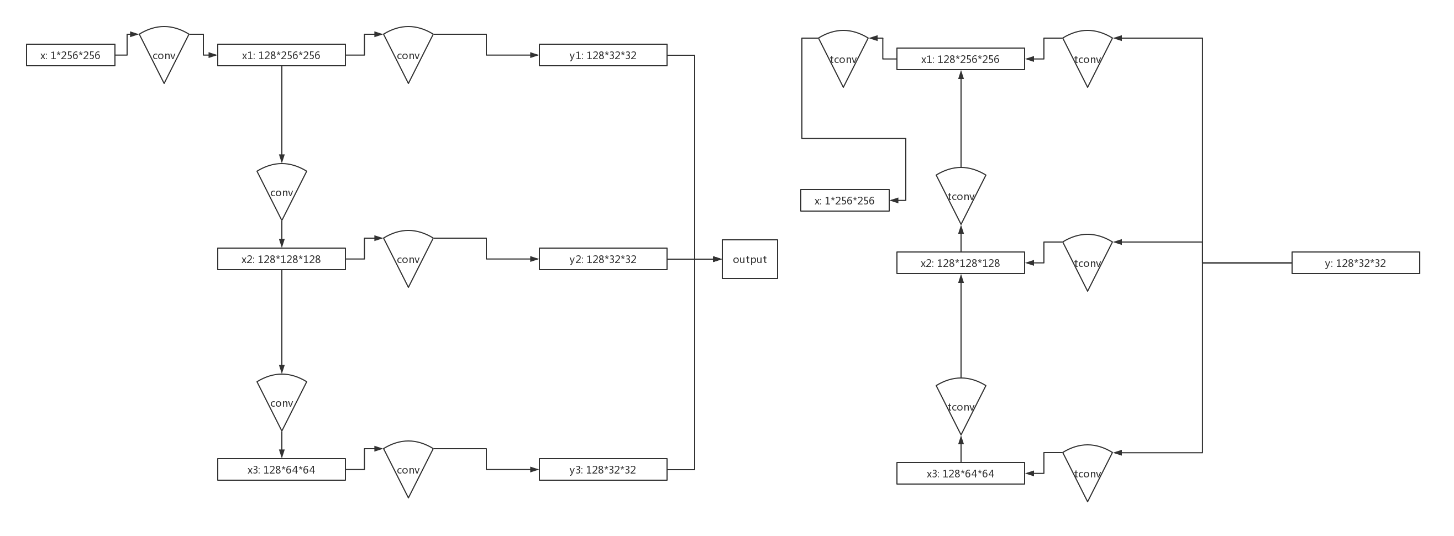
5\_4.py在横向的三角形conv前也加入了圆形conv层

5\_3\_3.py在5\_3.py的基础上，减小通道数进行测试

5\_3\_1.py在5\_3.py的基础上，输出降低为128\*16\*16

5\_3\_4.py 减小通道数到32

6\_1结构仍然如图，但输出是64\*32\*32：



6\_2改为32\*32\*32

6\_3改为64\*16\*16

6\_3\_1在6\_3基础上，输出增加一个不降维的全连接层

6\_3\_2改为100\*16\*16

6\_3\_3改为128\*16\*16

6\_4改为48\*32\*32

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 模型 | 编码器输出 | 相对尺寸/65536 | MSE | NLPL | MS-SSIM | 备注 |
| 6\_1 | 64\*32\*32 | 1 | 85.505 | 7.951 | 0.961 |  |
| 6\_2 | 32\*32\*32 | 0.5 | 402.444 | 21.170 | 0.868 |  |
| 6\_3 | 64\*16\*16 | 0.25 | 370.011 | 14.112 | 0.882 |  |
| 6\_3\_1 | 64\*16\*16 | 0.25 | 6580.907 | 25.730 | 0.234 | 在6\_3基础上，编码器输出增加一个全连接层 |
| 6\_3\_2 | 100\*16\*16 | 0.390625 | 377.226 | 13.988 | 0.899 |  |
| 6\_3\_3 | 128\*16\*16 | 0.5 | 277.429 | 11.197 | 0.932 |  |
| 6\_3\_4 | 64\*16\*16 | 0.25 | 659.527 | 25.343 | 0.735 | 在6\_3\_3的编码器输出后，多层卷积缓慢降低通道数 |
| 6\_3\_5 | 64\*16\*16 | 0.25 |  |  |  | 在6\_3的编码器中最低降到8\*8，然后升采样叠加到输出 |
| 6\_3\_6 | 32\*16\*16 | 0.125 |  |  |  |  |
| 6\_4 | 48\*32\*32 | 0.75 | 201.924 | 10.667 | 0.946 |  |
| 6\_5 | 256\*8\*8 | 0.25 | 1903.839 | 26.221 | 0.484 |  |
| 6\_6 | 128\*8\*8 | 0.125 |  |  |  |  |

7.py

结构如下所示，编码器输出为64\*16\*16

