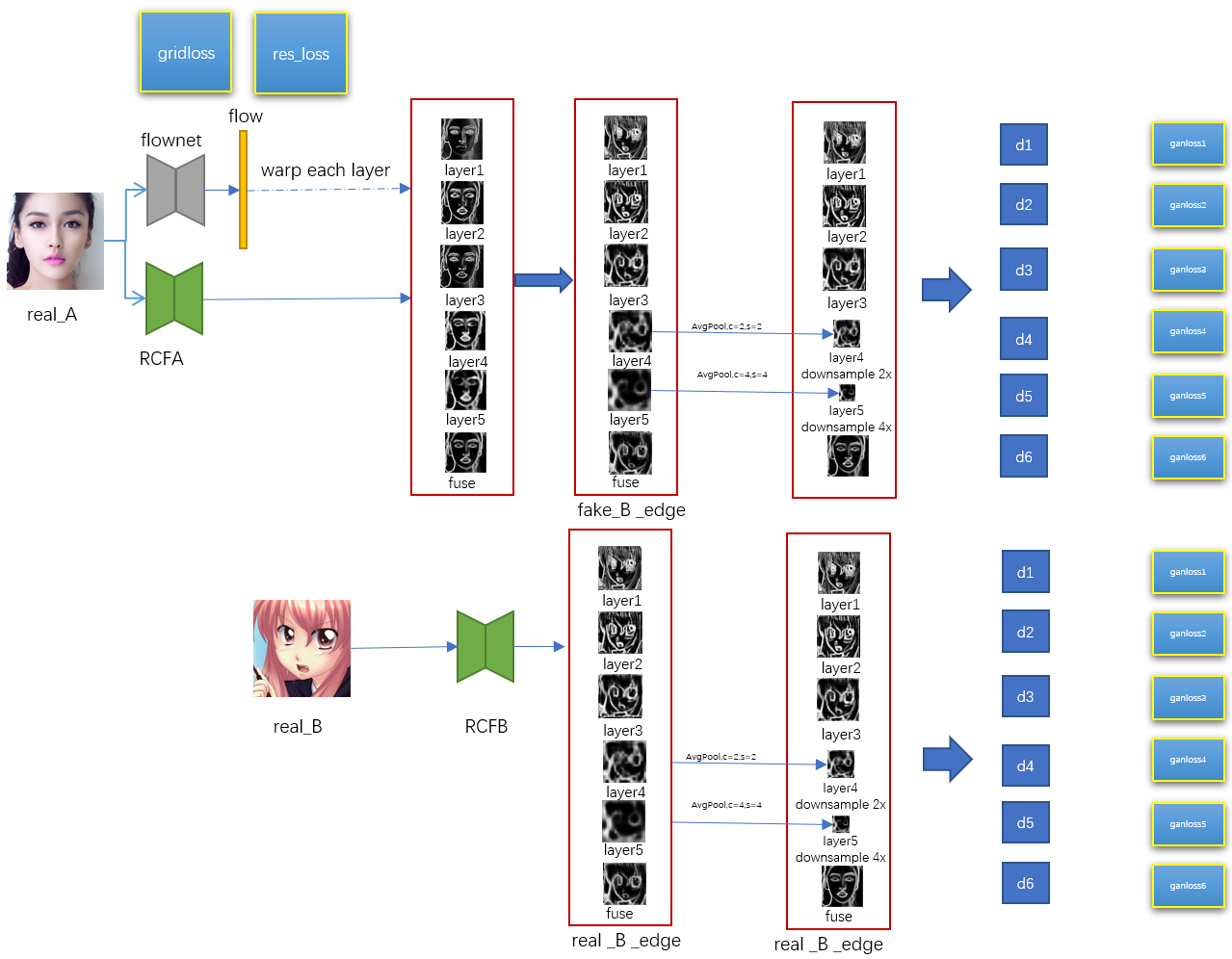
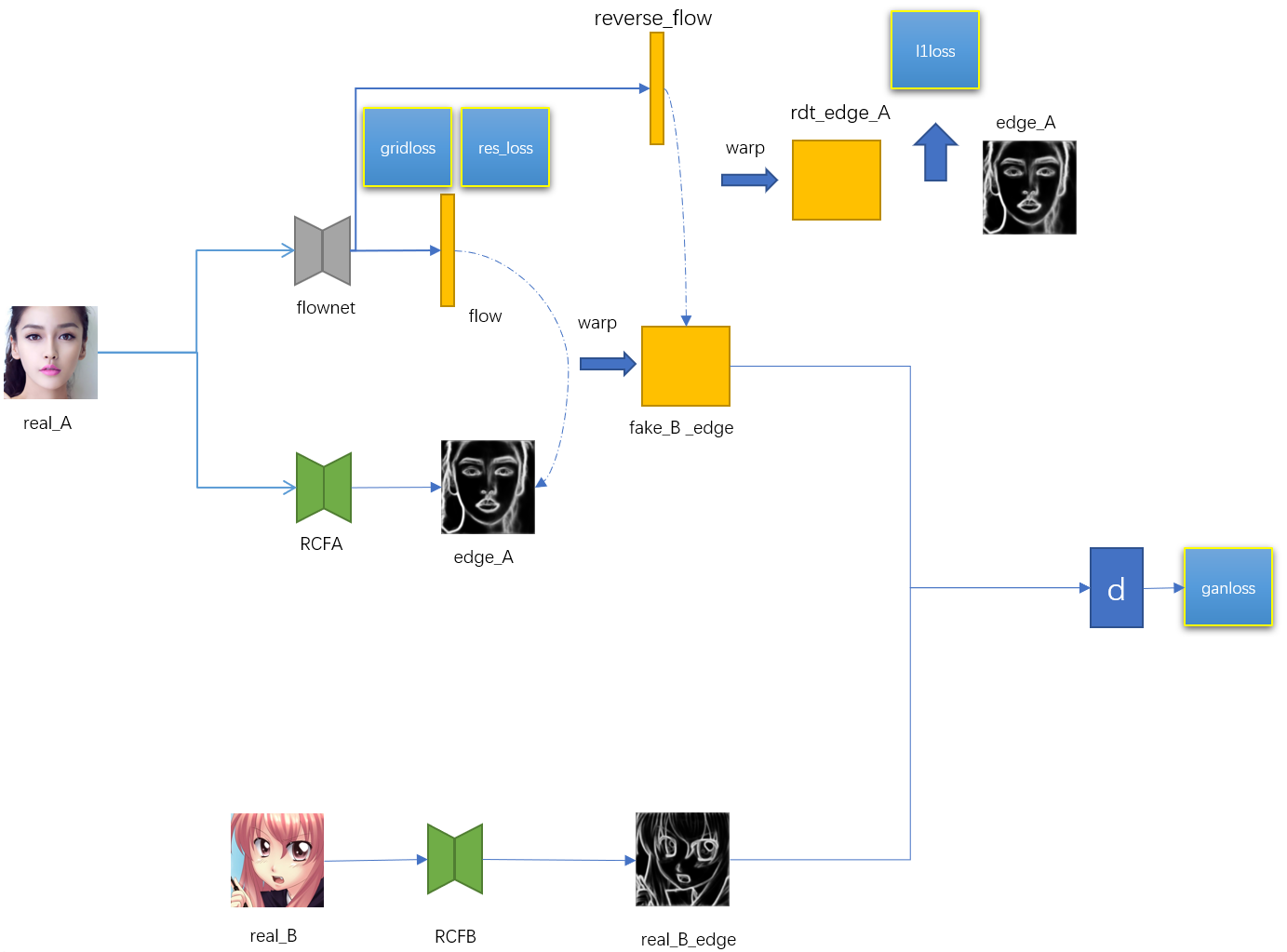
FlowNet实验



图a 带有多尺度判别器的网络



图b 带有reverse flow的网络

# **一、网络设置**

## **1.1 带有多尺度判别器的网络**

real\_A:真实人脸图片

real\_B:真实动漫图片

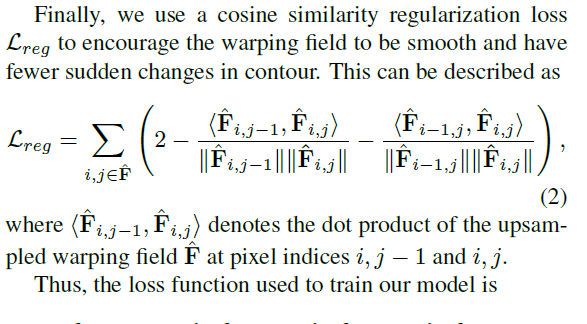
real\_B\_edge:真实动漫图片边缘

flow: FlowNet预测flow采用残差形式，FlowNet的输出是预测残差，

，

是图片像素点坐标。

grid\_loss:加在flow上的loss，目的是去掉一些不合理的warp。实现参照公式：



res\_loss:加在残差上的loss，目的是使得warp的程度尽可能小。

fake\_B\_edge:flow对real\_A的每层边缘进行warp后得到的边缘。

网络描述

首先real\_A经过RCF网络和FlowNet网络，分别获得RCF输出的每一层特征和flow，然后，用flow对RCF的每一层特征进行warp，目的是能得到动漫人脸的边缘轮廓，再对边缘特征的第四层、第五层的特征进行平均池化下采样，第四层是2倍下采样，第五次是四倍下采样。

同时real\_B经过RCF网络，得到六层特征，然后也使用对边缘特征的第四层、第五层的特征进行平均池化下采样。

为每一层都设置一个判别器，判别器使用的不是patch\_gan形式，而是预测一个数值(1/0)。

## **1.2 带有reverse部分的网络**

flownet预测输出flow和reverseflow。增加的reverseflow反过来warp fake\_B\_edge部分，得到rdt\_edge\_A，然后和edge\_A作l1loss。

# **二、数据扩增**

## **1、人脸动漫数据集**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **原图** | **layer0** | **layer1** | **layer2** | **layer3** | **layer4** | **refuse** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

rcf作用在真实人脸数据上

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **原图** | **layer0** | **layer1** | **layer2** | **layer3** | **layer4** | **refuse** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

rcf作用在动漫数据上

## **2、猫狗数据集**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **原图** | **layer0** | **layer1** | **layer2** | **layer3** | **layer4** | **refuse** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

rcf作用狗图片集

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **原图** | **layer0** | **layer1** | **layer2** | **layer3** | **layer4** | **refuse** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

rcf作用猫图片集

## **3、Webaricature数据集**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **原图** | **layer0** | **layer1** | **layer2** | **layer3** | **layer4** | **refuse** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

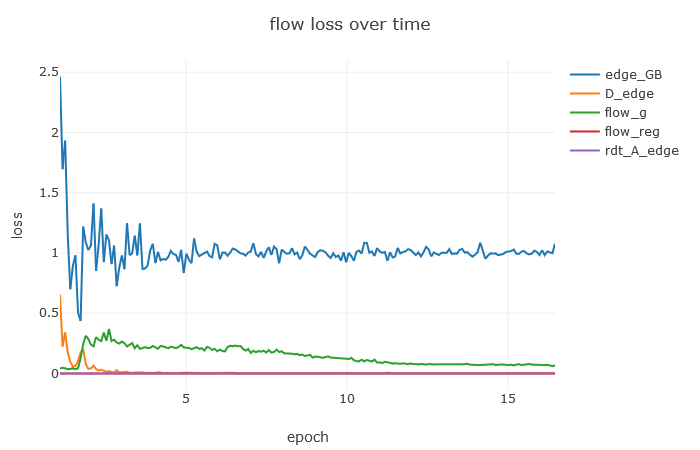
rcf作用Webaricature

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **原图** | **layer0** | **layer1** | **layer2** | **layer3** | **layer4** | **refuse** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

rcf作用Webaricature

## **实验**

首先在Webaricature数据上，使用图B网络使用loss如下



训练结果：随着训练的进行，发现warp的程度慢慢变小，生成器部分loss降不下来，判别器loss很低

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |