*FlowNet*部分实验安排以及进展

# 数据集

数据集来自两部分。

动漫数据采自seeprettyface\_anime\_face[1]，含有的噪声更少，采用了1539张。数据样例如图2.2

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

图2.2 动漫数据样例

第二部分是人脸图片，通过face++提取landmarks，同时归一化处理，采用了2002张。原图和处理之后的图如表2.1

|  |  |
| --- | --- |
| 原始图片 | 归一化后的图片 |
|  |  |
|  |  |
|  |  |

表2.1 真实人脸数据

# 网络结构以及*loss*设置

本次实验目的是通过*GAN*网络实现*FlowNet*，图片经过*FlowNet*生成*光流𝛷*，通过*𝛷*将图片的轮廓*warp*成，满足

其中代表提取边缘的函数，使用的是*RCF*网络，代表漫画图像的边缘空间。网络总体结构如图1.1

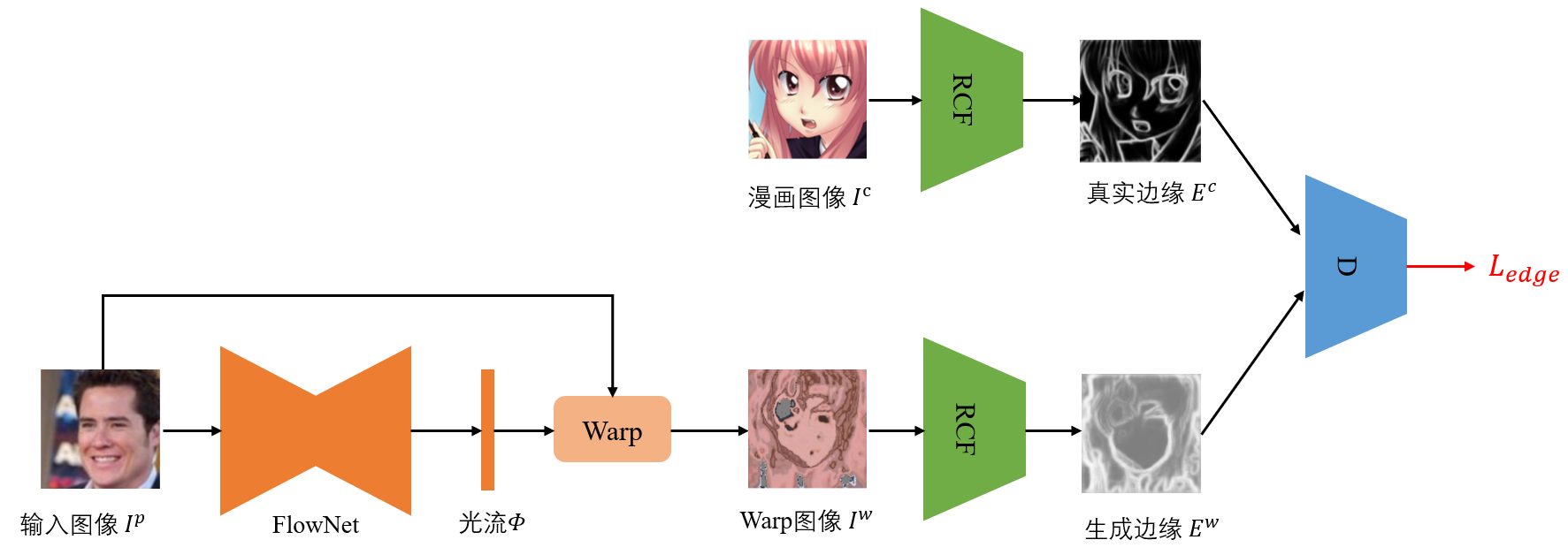


图1.1 训练*FlowNet*的总体网络结构

## 2.1 *FlowNet*的细节组成

*FlowNet*采用了*FlowNetSD*，网络结构如图2.1。

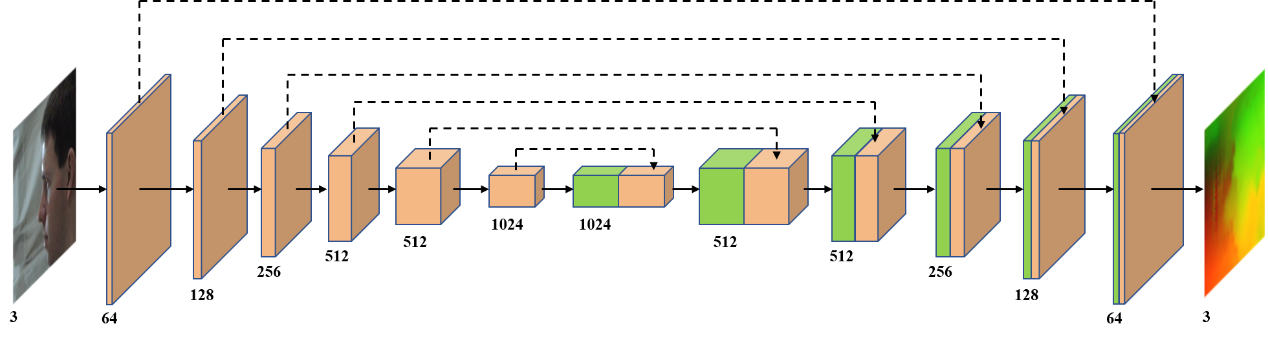


图2.1 *FlowNetSD*

## 2.2 *warp*公式

*warp*部分的本质是使用光流对输入图片进行采样，设的两个通道分别是、，在坐标的位置，两个通道分别取值、，如公式1.1

……………………………………(1.1)

那么输出图片位置的取值即为输入图片输入图片的取值，如公式1.2

…………………………………(1.2)

输入图片和*光流*经过warp部分处理后，得到，处理示意图如图1.5

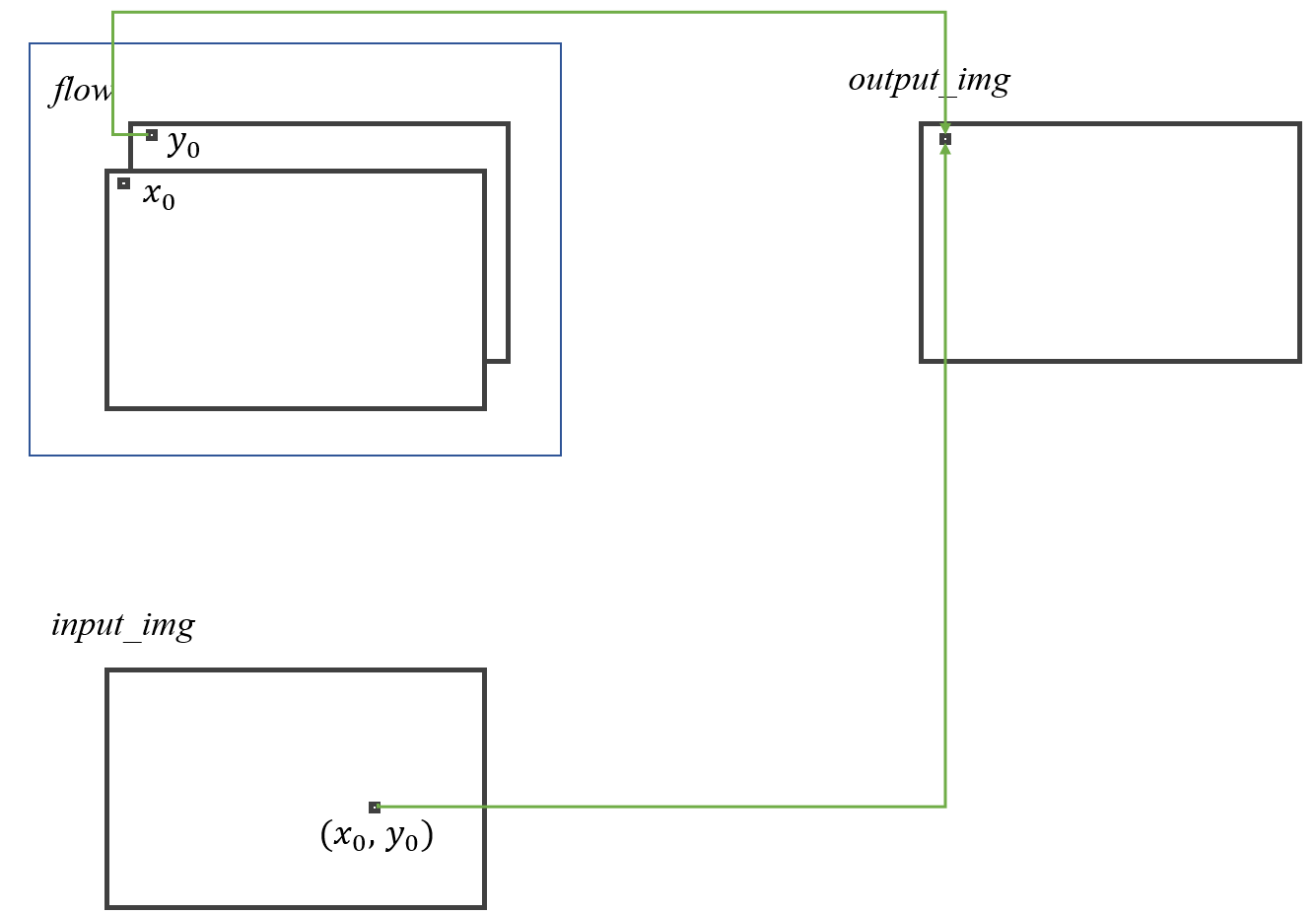


图1.5 *warp*过程

## 2.3 *RCF*网络

RCF网络使用原作者官方pytorch代码；输入，输出的为生成边缘；输入，输出真实边缘。网络结构图1.5，使用的是最后融合层*fuse*层

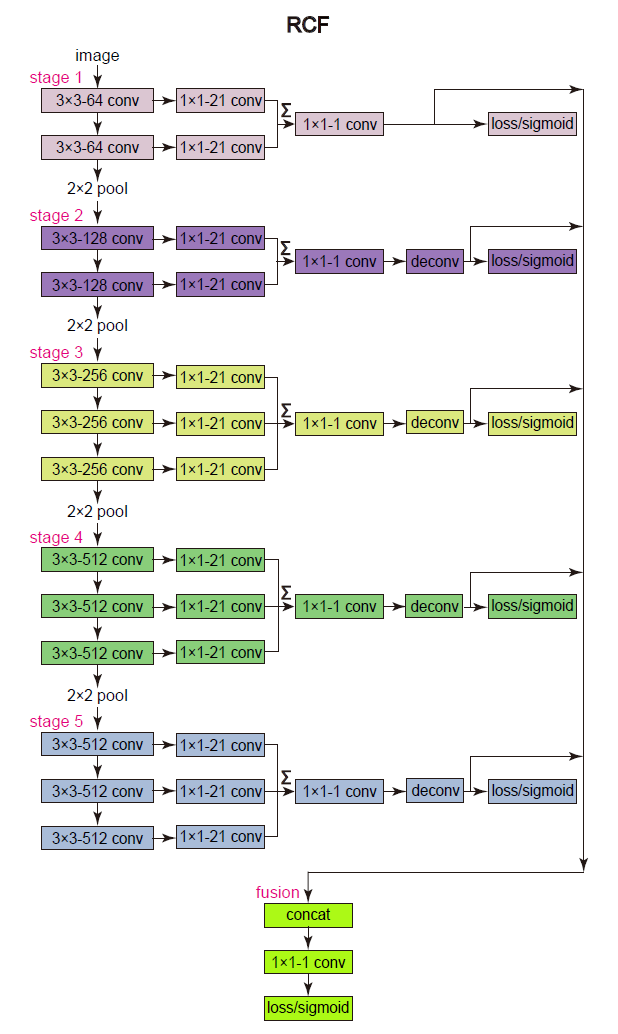


图1.5 RCF网络结构图

## 2.4 判别器*D\_edge*

判别器网络通过全卷积实现，进行64倍下采样，卷积核大小为4，然后生成一个和输出形状完全相同、所有维度都是1（判别为真）或者0（判别为假）的tensor，做MSEloss，的判别器网络图如图1.5

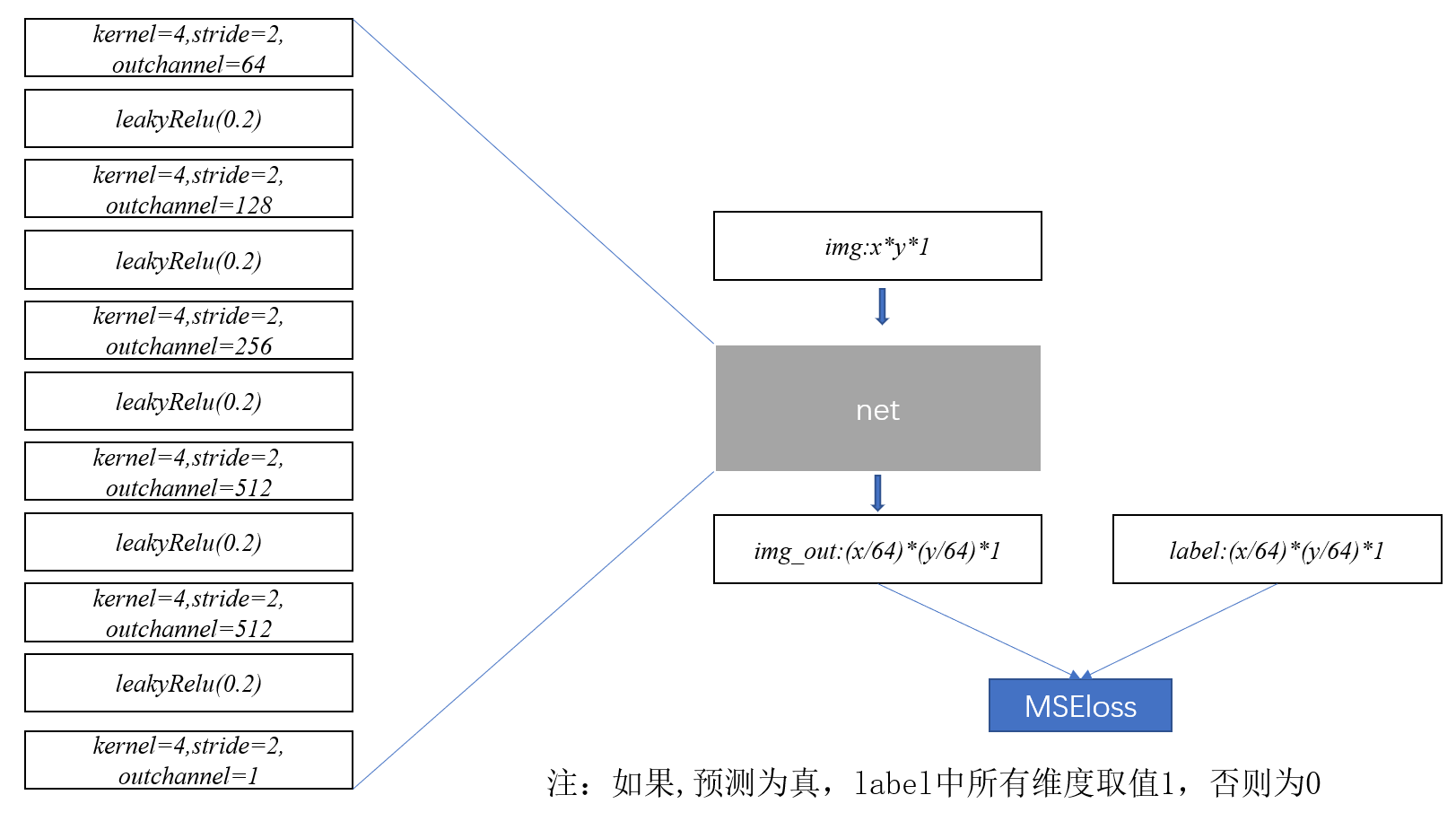


图1.5 判别器网络

## 2.5 *loss*组成

1. 判别loss

采用的公式如下,其中代表由漫画经过RCF网络生成的边缘；代表由Warp图像 经过RCF网络生成的边缘。此时训练的是判别器D\_edge的参数

1. 生成器loss

生成器loss训练的参数是*FlowNet*中的参数

1. TVloss

将加在*光流*上，目的是使warp的程度尽量较小。的公式如下

是一个的张量，其中，对应位置上的[*b, c, h, w*]代表第b张图的(h, w)的c（x或者y）方向上的位置

# 实验

## 3.1不加入

这次实验的loss中，生成器和判别器都只使用GANloss，每次batch为1张，一共训练了40个epoch，loss波动如图3.1，蓝色代表生成器flowloss，橙色代表判别器判别器loss。

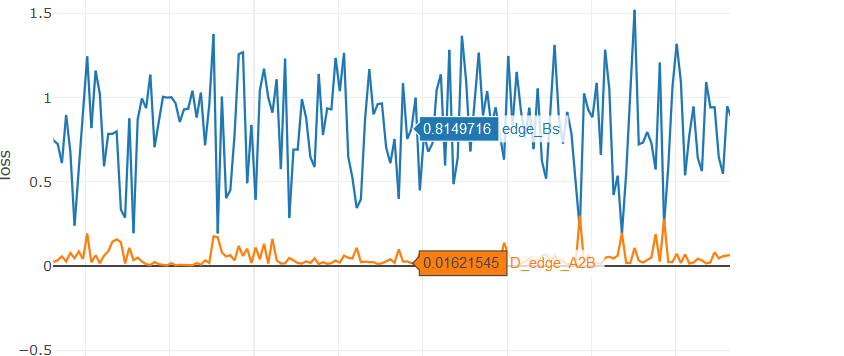


图3.1 训练loss波动

实验结果

|  |  |  |  |
| --- | --- | --- | --- |
| epoch | real\_A | Bs | fake\_edge |
| 1 |  |  |  |
| 5 |  |  |  |
| 10 |  |  |  |
| 20 |  |  |  |
| 30 |  |  |  |
| 40 |  |  |  |

## 3.1加入

本次实验进行了30个epoch，实验结果如下。loss如图3.2

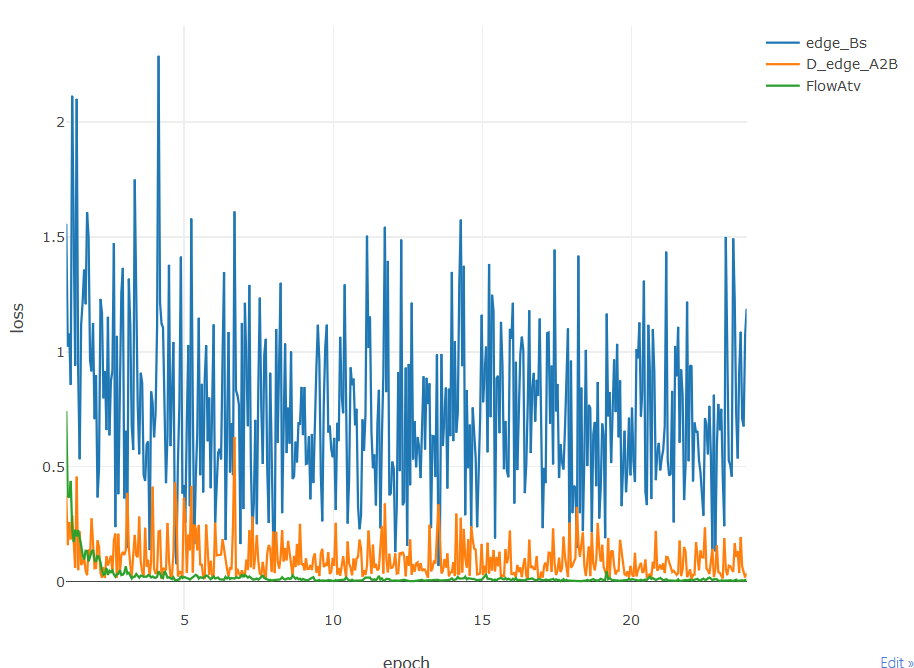


图3.2 加入之后的训练*loss*

实验效果图

|  |  |  |  |
| --- | --- | --- | --- |
| epoch | real\_A | Bs | fake\_edge |
| 1 |  |  |  |
| 5 |  |  |  |
| 10 |  |  |  |
| 15 |  |  |  |
| 20 |  |  |  |

# 附录

[1] http://www.seeprettyface.com/mydataset\_page3.html#mulu