pix2pix技术报告

**学号 19063140 姓名 郑凯心**

摘 要 这篇论文主要讲了如何将conditional GANs运用到Image-to-Image任务中，并且讲了如何设计网络来取得很好的效果。主要有两点：对于各种不同的问题（论文指的应该是Image-to-Image任务）conditional GANs都能给出比较好的结果；论文提出了一种简单的网络框架能够得到比较好的结果，同时论文中分析了几种结构的影响。

关键词 pix2pix cGAN PatchGAN U-Net

# **1** 介绍

图像处理的很多问题都是将一张输入的图片转变为一张对应的输出图片，比如灰度图、梯度图、彩色图之间的转换等。通常每一种问题都使用特定的算法（如：使用CNN来解决图像转换问题时，要根据每个问题设定一个特定的loss function 来让CNN去优化，而一般的方法都是训练CNN去缩小输入跟输出的欧氏距离，但这样通常会得到比较模糊的输出）。这些方法的本质其实都是从像素到像素的映射。于是论文在GAN的基础上提出一个通用的方法：pix2pix 来解决这一类问题。通过pix2pix来完成成对的图像转换(Labels to Street Scene, Aerial to Map,Day to Night等)，可以得到比较清晰的结果。

# **2** 相关工作

## **2.1** Structured losses for image modeling

Image-to-Image转换问题常常被当作是逐像素的分类或者是回归问题，而这些构想的输出往往没有将结构考虑在内，输出的像素被当做是与输入图片的其他像素条件无关的。但是GAN却能学习到结构损失（structures loss）。

## **2.3** Conditional GANs

作者在这篇论文中使用了Conditional GANs，但是在结构上与之前的工作不同。在生成器中作者使用了“U-Net”-based的结构，在判别器中使用了convolutional “PatchGAN”分类器，这种分类器只是惩罚images patches范围内的结构损失。

# **3** 方法

训练大致过程如上图所示。图片 x 作为此cGAN的条件，需要输入到G和D中。G的输入是{x,z}（其中，x 是需要转换的图片，z 是随机噪声），输出是生成的图片G(x,z)。D则需要分辨出{x,G(x,z)}和{x,y}。

## **3.1** 目标函数

一般的cGAN的目标函数如下，生成器 G 不断的尝试minimize下面的目标函数，而D则通过不断的迭代去maximize这个目标函数。

为了测试输入的条件x对于D的影响，论文也训练一个普通的GAN，判别器D只用于判别生成的图像是否真实。

前人的一些工作中发现，将GAN的目标函数和传统的loss结合，可以带来更好的效果。所以论文增加了一个L1 loss交给生成器G去最小化。

所以最终的目标函数是：

文中对于不同的loss的效果做了一个对比，可以看到L1 + cGAN的效果相对于只用L1或者cGAN都是比较好的。

## **3.2** 网络结构

论文对DCGAN的生成器和判别器的结构做了一些改进。

### 3.2.1 生成器结构

U-Net是德国Freiburg大学模式识别和图像处理组提出的一种全卷积结构。和常见的先降采样到低维度，再升采样到原始分辨率的编解码(Encoder-Decoder)结构的网络相比，U-Net的区别是加入skip-connection，对应的feature maps和decode之后的同样大小的feature maps按通道拼(concatenate)一起，用来保留不同分辨率下像素级的细节信息。U-Net对提升细节的效果非常明显，不同尺度的信息都得到了很好的保留。

### 3.2.2 判别器结构

利用马尔科夫性的判别器(PatchGAN)

pix2pix采用的一个想法是，用重建来解决低频成分，用GAN来解决高频成分。一方面，使用传统的L1 loss来让生成的图片跟训练的图片尽量相似，用GAN来构建高频部分的细节。

另一方面，使用PatchGAN来判别是否是生成的图片。PatchGAN的思想是，既然GAN只用于构建高频信息，那么就不需要将整张图片输入到判别器中，让判别器对图像的每个大小为N x N的patch做真假判别就可以了。因为不同的patch之间可以认为是相互独立的。pix2pix对一张图片切割成不同的N x N大小的patch，判别器对每一个patch做真假判别，将一张图片所有patch的结果取平均作为最终的判别器输出。

具体实现的时候，作者使用的是一个NxN输入的全卷积小网络，最后一层每个像素过sigmoid输出为真的概率，然后用BCEloss计算得到最终loss。这样做的好处是因为输入的维度大大降低，所以参数量少，运算速度也比直接输入一张快，并且可以计算任意大小的图。论文对比了不同大小patch的结果，对于256x256的输入，patch大小在70x70的时候，从视觉上看结果就和直接把整张图片作为判别器输入没有多大区别了：

# **4** 实验结果

Table1是关于不同损失函数的组成效果对比， 这里采用的是基于分割标签得到图像的任务。评价时候采用语义分割算法FCN对生成器得到的合成图像做语义分割得到分割图，假如合成图像足够真实，那么分割结果也会更接近真实图像的分割结果，分割结果的评价主要采用语义分割中常用的基于像素点的准确率和IOU等。

Table2是关于不同生成器的效果，主要是encoder-decoder和U-Net的对比。

Table3是关于判别器PatchGAN采用不同大小N的实验结果，其中1 ∗ 1 1\*11∗1表示以像素点为单位判断真假，显然这样的判断缺少足够的信息，因此效果不好；286 ∗ 286 286\*286286∗286表示常规的以图像为单位判断真假，这是比较常规的做法，从实验来看效果也一般。中间2行是介于前两者之间的PatchGAN的效果，可以看到基于区域来判断真假效果较好。

# **5** 总结

优点：pix2pix巧妙的利用了GAN的框架来为“Image-to-Image translation”的一类问题提供了通用框架。利用U-Net提升细节，并且利用PatchGAN来处理图像的高频部分。

缺点：训练需要大量的成对图片，比如白天转黑夜，则需要大量的同一个地方的白天和黑夜的照片。

参 考 文 献

1. Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Networks, <https://arxiv.org/pdf/1611.07004.pdf>
2. 砖业人士, [GAN笔记] pix2pix, <https://www.jianshu.com/p/8c7a7cb7198c>
3. ZonyH, 论文阅读——《Image-to-Image Translation with Conditional Adversarial Networks》, <https://blog.csdn.net/zh20166666/article/details/83651519>
4. AI之路, pix2pix算法笔记, https://blog.csdn.net/u014380165/article/details/98453672